1. Executive Summary

This strategy covers how the Council will adopt Digital and Agile methods to improve its services and the products that underpin those services in conjunction with the council's ambitions.

The document is intended for anyone who is considering replacing or making a significant change to a system or, process that is, or could be, supported by technology. It is meant as a guideline to the standards and principles to be adopted when developing services in a digital way and supports the Digital Charter which, the Authority is a signatory.

Digital ways of working change the way we think about the users of our services; to enable them to easily interact with the systems and processes in place to support the delivery of services and information. It is not about technology which, is a facilitator of Digital ways of working, rather it is changing the way we develop our services to embrace ways of working that make interaction with the council easier while ensuring accessibility is maintained.

The current suite of applications has, in the main, been developed using traditional methods of waterfall development and project approaches. While these systems have derived benefit their development has resulted in systems that are frequently sub-optimal and that create a myriad of work arounds that are "off system" to accommodate inefficiencies and problems created through systems implementation and operation.

To adopt Digital working practices and comply with the Digital Declaration which, the council has signed up to, then there are principles and processes that need to be adopted as part of the governance of Digital work. This strategy outlines what these are and what the council will adopt to enable it to become a Digital organisation.

As Digital and Agile ways of working are adopted the need for a separate Digital strategy should disappear as any project or, programme-based work, will adopt this way of working as a matter of course. Overtime there will be no need to preface any particular piece of work with the word "Digital" as this will become second nature and it will be the way the council works when developing services or systems.

The ICT strategy will form a key foundation for the Digital strategy as it will provide the technology guiderails upon which services will be developed. The ICT strategy will also highlight areas of technical debt that need to be addressed prior to a significant system or, service upgrade/implementation otherwise there would be a risk of developing services that are immediately unsustainable.

2. Demand

2.1 Business Context

The council has the stated aim of being “Ambitious for our future” which includes, “A forward-looking and resilient Council”. This was expanded further, “Our staff stepped up during the pandemic and worked hard to adapt services and protect the most vulnerable. We want to continue to support them by giving them the skills and knowledge they need and providing a positive, inclusive culture. We'll also continue to develop new and innovative ways of working, including some which came out of the pandemic, like hybrid home/office working, and using 'virtual visits' to connect social care staff and residents. We're always looking to make services more efficient and sustainable and better meet people’s needs, and will carry on looking at ways to develop our services and change the way we work in the future”

The Applications that underpin all the processes within the council sit on the Council’s ICT infrastructure.

Currently most of those Applications, Systems and Processes, that are supported within the council have been developed in traditional ways; that is gathering business requirements and commissioning a provider (internal or external), to configure and implement a solution with little or no user involvement throughout the development process. This has meant that systems were implemented that did not always meet the expectations of the users. A number of these systems are now aged, life expired or, becoming unsupportable and creating process bottlenecks plus, significant technical debt. This gives us an opportunity to rethink how we approach developing these systems and processes.

The Council has started to adopt Agile ways of working which, has brought benefits in the products developed, in that they are developed in conjunction with the users of those systems with user feedback and prioritisation of features. This results in products that are performant plus, they match users expectations. An example is the Brokerage in Adults which, the Mosaic software supplier recognizes as best practise. This approach to work now needs to be scaled, but not at the expense of a working quality product which means, prioritising this activity across the council to facilitate the allocation of sufficient resource.

This can only be achieved by a change in the way we develop services and the underpinning applications and systems for the future. To this end Digital methods will be adopted in conjunction with Agile ways of working to support user centred design and develop and/or replace systems and processes from end to end, so moving towards more integrated interactions with its citizens and staff. This way of working will support the Authorities commitment to the Digital Declaration.

The council also has several transformation programmes and the work to underpin those programmes will need to be prioritised across the council, taking into consideration the availability of resources.

This strategy will seek to provide the guidelines within which, Digital approaches can be adopted by utilising the current debt pile of life expired systems and technical debt, not just to prioritise replacement of like with like systems, but to take the opportunity, wherever possible, to rethink how those processes work from an end-to-end perspective. Putting the user at the heart of the design of those systems and services while integrating with partners where that makes the user experience better.

We will also look to utilise technologies to capture data in an ethical manner to help interventions which support our service users and minimise the disruption of their lives. Where appropriate this technology will automatically update systems to provide a holistic view of our service users and trigger helpful interventions to support citizens.

The combination of looking at systems and services from an end-to-end perspective and automating the interactions across the council will also drive out efficiencies in the way the council functions and enable professional staff to focus on their users.

2.2 Business Success

To meet the council's ambitions, digital services will aim to achieve a single user contact, to initiate delivery of the service or, information to that user, with all the complexity across the council and it partners automated and the user interface being intuitive and easy to use.

If the service being requested is transactional then this should be online as the primary method of interaction however, currently most are not intuitive and many involve phone calls or, multiple interactions with people plus several hand offs between functions/ business areas and repeat requests for the same or, similar information. A good current example of this would be colleagues who which to transition gender. Currently the systems and processes in place to alter all references to an employee to reference the new gender make this challenging and frustrating for those colleagues transitioning, such that we are seeking ways to make this administrative process simpler so removing an unnecessary stress.

There is also significant benefit to be derived through the ethical use of data and technologies that can support the monitoring and interaction of users in a way that prevents injury or harm plus, can promote proactive interaction for better wellbeing. An example of this would technology deployed in people's home to prevent falls and monitor activity so that people can continue to live at home safely. As we progress through our journey of using more sophisticated toolsets on our data this will start to reveal ways in which that data can be interrogated for the benefit of our users.

Of course, to achieve that ability there is a need to map the current data landscape plus, give an understanding of the ownership of the data across the council and its mastery. Who owns the data and the primary source of data will need to be quantified to achieve the benefits of any new technologies that are deployed. This will also enable "store once use many times" to prevent inconsistencies and support better reporting.

This activity will be informed by the Data Strategy work which will set out the principles by which the Authority wants to govern and share data.

2.3 Business Capabilities

To achieve Digital transformation the traditional ways of developing systems and services need bolstering to support the principle of user centered design. Digital and Agile methods will need to be deployed to support the necessary change in the way systems and services are developed, configured, and implemented. These methods demand greater input from the users of the system and concentrate on, 1) Business Value and 2) Usability. There are other factors however, these are the two over-riding aspects. This means different business capabilities are required to enable Digital services to be developed.

Those capabilities are:-

A move to using the artefacts produced from Digital and Agile methods. That means that the traditional hierarchical approach being turned on its head. The project structure looks to support the delivery rather than being a control/reporting mechanism. Projects with large ICT elements traditionally are difficult to quantify therefore they overrun and do not deliver what was expected. To prevent this the ownership needs to be across the governance structure, managing the uncertainty and the blockers that exist within these types of projects and making timely decisions to maintain the cadence of the delivery. This means working from different artifacts such as, *backlogs, burn downs and burn up charts to manage delivery as opposed to Highlight reports. These are new ways of working and will require learning and embedding. Cost control and planning are integral parts of delivery, and this way of working gives significantly more visibility than traditional methods however, it demands quick decision making to address blockers to progress. This means the traditional board structure will not work because it is not frequent enough; it can supplement Agile approaches as an additional governance layer to give oversight but, is not enough in itself.

There are specific roles necessary to the successful delivery of Agile and Digital Projects:
User Researcher (UR) – someone who works with the User community to establish how a system and/or process should work to make the users experience seamless and straight-forward such that digital interaction becomes the way of doing business.

User eXperience (UX) – The front-end interface and usability to make sure the user experience one the device and transaction being undertaken is seamless – this can be combined with the User Researcher where an Application is being deployed.

Service Owner (SO)- someone who owns the end-to-end service and can make priority decisions on how the service develops plus, support any business case development, associated resourcing and, negotiation around process ownership.

Service Designer – would typically map out the entire service design and work with the Service Owners, Product Owners and User Researchers and the business areas to design the service from an End-to-End perspective. If a package solution is being procured with built in workflows then the Product Owner can work with the Service Owner to achieve the same outcome.

Product Owner (PO)- someone who takes ownership of the product that will be delivered to the Service Owner. They work with the Scrum team(s) daily and set the priorities. To do this the Service Manager needs to delegate the responsibility for the daily decision making that keep the team delivering. These two roles (Service and Product Owners) are key to success as they determine the priority in terms of what is developed and are key to unblocking any issues that are hampering delivery.

Scrum Masters (SM) and Developers (Devs) are two other capabilities that are critical to success, the Scrum master works hand in glove with the product owner to manage the backlog and seeks to unblock elements that prevent the developers from progressing. It is helpful if the scrum master comes from a technical background but not critical.

Developers will configure and write the code to build the systems and integrations however, their level of expertise varies depending on what is being implemented package, Software as a Service (SaaS), Bespoke development. They are critical to the success of any implementation.

For the council to be successful in producing digital services there will be a need to have these capabilities available at the appropriate times to support timely delivery. Without the capabilities itemized above then delivery will be hampered and is likely to be sub-optimal.

There is also a significant requirement for skills that do exist within the Authority but possibly not in the volumes necessary such as Business Analysts, Technical Architects, and Developers. These will need to be bolstered to support this way of working.

In addition, the governance structures need to alter from being predominantly about command and control to being about facilitating delivery and oversight. This changes the dynamic of the participants and will require a cultural change in approach.

*See Glossary

2.4 IT Contribution to Business Success

Currently, in the main, services are developed in a very traditional way whereby a specification is produced at the beginning of a project and the supplier, internal or external, takes that document and builds what they think is meant by that specification. Project reporting happens on a regular basis outlining progress however, the visibility of what is being produced is generally opaque until testing, at which point what is delivered does not usually reflect what the user expectation was. Also, the level of user engagement and ownership is limited, and this can drive an unhelpful working relationship which, becomes contractual and confrontational. This does not help the delivery of a useable, elegant product to our users; indeed, it can aggravate all parties and normally ends up in sub-optimal products being deployed.

By using Digital and Agile methods of delivery there is a need to have an input and ownership from across the user / supplier divide. The Service and Product Owners are key to the success of this way of working as they see what is being produced and understand the business and can make decisions quickly on how systems and processes should be developed and configured. This means that the governance needs to support the Service and Product owners by empowering them to make decisions on a day-to-day basis. In addition, the traditional Board structure needs to adapt to the ceremonies and artefacts that are produced from Digital and Agile ways of working. This means attending stand-ups and having a different role; rather than overseeing they will be unblocking items that are preventing delivery. They will also need to respond at a much faster cadence than traditional boards operate, and this will take some time to achieve this cultural change.

The benefit is simple, the products and services produced are easy to use for our citizens and staff plus, they are joined up across service and partner boundaries, removing artificial organisational constructs and making our services more efficient and effective.

3. Control

3.1 Digital Principles

The UK Government have a set of 10 digital principles which are detailed below. These apply to government services when they are under development. Therefore, it is appropriate that the Authority adopts these given they are in line with current government thinking and are underpinned by Agile principles (ANNEX 1). They are then checked at specific points by an assessor using the GDS Service Standards. The Digital principles are as follows and each one includes links as examples and to support the principles:

1) Start with User Needs

Service design starts with identifying user needs. If you don't know what the user needs are, you won't build the right thing. Do research, analyse data, talk to users. Don't make assumptions. Have empathy for users and remember that what they ask for isn't always what they need.

- Why we care more about effectiveness than efficiency or satisfaction by John Waterworth
- 10 tips for working with your user researcher by Kieron Kirkland

2) Do less

Government should only do what only government can do. If we've found a way of doing something that works, we should make it reusable and shareable instead of reinventing the wheel every time. This means building platforms and registers others can build upon, providing resources (like APIs) that others can use, and linking to the work of others. We should concentrate on the irreducible core.

- Adapting the GOV.UK Design System for the NHS by Tim Paul and Dean Vipond
- GOV.UK: a journey in scaling agile by Jen Allum, Nick Tait and Alan Wright

3) Design with Data

In most cases, we can learn from real world behaviour by looking at how existing services are used. Let data drive decision-making, not hunches or guesswork. Keep doing that after taking your service live, prototyping and testing with users then iterating in response. Analytics should be built-in, always on and easy to read. They're an essential tool.

- Mixing methods: a recipe for research success by Haur Kang and Louise Petre
- How to use data in user research when you have no web analytics by Louise Petre

4) Do the hard work to make it simple

Making something look simple is easy. Making something simple to use is much harder - especially when the underlying systems are complex - but that's what we should be doing.

Don’t take “It’s always been that way” for an answer. It’s usually more and harder work to make things simple, but it’s the right thing to do.

- Building a better GOV.UK, step by step by Gabrielle Acosta and Sam Dub
- Start taking payments in one day with GOV.UK Pay by Katie Bates, Lewis Dunne and Till Wirth

5) Iterate and then iterate again

The best way to build good services is to start small and iterate wildly. Release minimum viable products early, test them with actual users, move from alpha to beta to live adding features, deleting things that don’t work and making refinements based on feedback. Iteration reduces risk. It makes big failures unlikely and turns small failures into lessons. If a prototype isn’t working, don’t be afraid to scrap it and start again.

- Things to consider when designing in alpha by Chris Thomas
- Using A/B testing to measurably improve common user journeys by Mark Mcleod

6) This is for everyone

Accessible design is good design. Everything we build should be as inclusive, legible, and readable as possible. If we must sacrifice elegance - so be it. We’re building for needs, not audiences. We’re designing for the whole country, not just the ones who are used to using the web. The people who most need our services are often the people who find them hardest to use. Let’s think about those people from the start.

- Using persona profiles to test accessibility by Anika Henke
- What we found when we tested tools on the world’s least accessible webpage by Mehmet Duran

7) Understand context

We’re not designing for a screen, we’re designing for people. We need to think hard about the context in which they’re using our services. Are they in a library? Are they on a phone? Are they only really familiar with Facebook? Have they never used the web before?

- Researching in context by Amy Everett
- Hey GOV.UK, what are you doing about voice? by Sam Dub and Mark Hurrell

8) Build digital services, not websites

A service is something that helps people to do something. Our job is to uncover user needs and build the service that meets those needs. Of course, much of that will be pages on the web, but we’re not here to build websites. The digital world must connect to the real world, so we have to think about all aspects of a service, and make sure they add up to something that meets user needs.

- What do we mean when we talk about services? by Stephanie Marsh
- How we approached service mapping by Gabrielle Acosta and James Butler

9) Be consistent, not uniform

We should use the same language and the same design patterns wherever possible. This helps people get familiar with our services, but when this is not possible, we should make sure our approach is consistent.

This isn’t a straitjacket or a rule book. Every circumstance is different. When we find patterns that work, we should share them, and talk about why we use them. But that shouldn’t stop us from improving or changing them in the future when we find better ways of doing things or the needs of users change.

- Introducing a simpler, data-informed publishing experience by Rob Rockstroh
- Introducing the GOV.UK Design System by Amy Hupe and Alice Noakes

10) Make things open: it makes things better

We should share what we’re doing whenever we can. With colleagues, with users, with the world. Share code, share designs, share ideas, share intentions, share failures. The more eyes there are on a service the better it gets - howlers are spotted, better alternatives are pointed out, the bar is raised.

Much of what we’re doing is only possible because of open-source code and the generosity of the web design community. We should pay that back.

- Opening up the GOV.UK Design System for contributions by Ignacia Orellana and Amy Hupe
- Doing the hard work to make things open by Paul Smith

3.2 Digital Governance

The control mechanisms are outlined in the benefits above. For Digital and Agile methods to work there is a need to alter the governance to support these ways of working and empower staff to make local, quick decisions. As defined in Agile principles “Business people and developers must work together daily throughout the project”

For the governance to be successful the board needs to be able to react faster than a traditional board and will need to adapt to accommodate this way of working. In addition, the board becomes a facilitator of the project/programme, so while it has its traditional governance hat on for oversight, it is also accountable for removing any blockers to progress.

This means that the stakeholders will need to embrace the Agile methodology and attend retrospectives and “Show & Tell” sessions to gain insight into what is being developed. To delegate this to other officers does not offset accountability for the product and/or service development as that will be implicit within the role of the senior stakeholder(s).

Agile shall be the preferred method of delivery for digital projects and programmes, recognizing that this will need to be modified depending on the type of delivery and the phases are recognized by Government Digital Services (GDS) will be adopted. Those being Discovery, Alpha, Private Beta, Public Beta, and Live. This together with the GDS Service Standards (ANNEX 2), will support and guide the development of systems and services in a way that is supports the production of good customer focused services.

It should also be recognized that Minimum Viable Service (MVS) is just that, and it is likely that an equal amount of time and resource will be required after that point to reach a fully mature service.

In terms of planning a service the governance constructs will need to accommodate, the development lifecycle of the service. Once MVS is reached there will need to be a careful management of resource to enable the parallel activities of Business as Usual (BAU) and development to take place. This should form one backlog and the Product and Service Owners will need to prioritise carefully sustainability and new features otherwise significant technical debt will be created and the service is likely to be unreliable.

3.3 IT Financial Management

Each Product and/or Service will have to produce their own business case to justify the expense of any new delivery that demands additional resource above that which is already allocated. Some developments may fall within existing resources agreed at the beginning of the financial year and these will be agreed at the Digital Board as this will be the mechanism of prioritisation.

The benefits

Each year the prioritisation will be undertaken to determine what work can be contained under existing resources, following that exercise additional activity will need to provide a business case to justify the expenditure. The business case will need to cover the ongoing running costs as well as the direct project costs.

4.0 Enterprise Architecture

This Digital strategy will set the principles and identify a target ICT Technical Architecture for 3 years from 2021. It is split into different areas of the infrastructure and will define a direction based on information available at the date of writing along with any longer-term predictions by the industry while taking into account the strategic direction Nottinghamshire County Council in order to support the wider authority in delivering efficiencies and help the move to digital.

ICT Achievements 2020

Within the difficult year of 2020 and early 2021 ICT have enabled and expanded a number of the Infrastructure technologies to help ensure the Authority could continue to deliver its services while maintaining the safety of the workforce. The main elements of which were the major work done around delivery of the Microsoft M365 Office and Teams products along with the network complexities to support the working from home environment ensuring the workforce could access their usual applications. These along with setting up further elements for Cloud Services have built a strong foundation for the future Architecture.

Architecture Elements

The following elements of the ICT Architecture ICT strategy will be covered:

Compute Architecture
This includes any physical and cloud-based devices used for processing and storage along with the network components for the delivery of data and voice communications within the Authority and beyond.

Storage
This includes the elements that make up an application both within the cloud and on-premise but not the individual applications functionality. Typically, this consists of a database with an application along with an access element, usually a PC or Web browser.

Network Architecture

This includes any physical and cloud-based devices used for processing and storage along with the network components for the delivery of data and voice communications within the Authority and beyond.

Microsoft Cloud Services M365
As we move to cloud based services more of our information will be stored in the cloud. Along with the Microsoft 365 suite and associated services we will need to ensure this covers our requirements and take advantage of opportunities to leverage the technology to make efficiencies in the organisation.

Security Architecture
The Security Architecture and policies allow us the keep our information safe and ensure none authorised access is reduced and monitored. The Security Architecture is built to industry standards and is checked yearly to identify any weaknesses and ensure we meet those standards.

Access Channels and Devices
This area covers the mechanisms and devices used to access our systems and data whether on the local network, remotely from home or out and about.

Data
A separate Data Strategy is currently being constructed. Once that has been agreed the physical implementation of that will be considered alongside the strategy. However, the general principle of data being entered once and used many times will apply and be reflected in any principles created.

A more detailed document that expands on most of the elements above is included in Annex (3).

4.1 People

The capabilities listed above in section 2.3, only exist in limited numbers within the Authority. There will be some significant training needs that will need to be addressed and it is proposed that a training needs analysis is undertaken at the start of any project to highlight those costs.

The costs of the training would then be incorporated into the business case for the service or system that is being developed.

4.2 Sourcing

It is likely that additional resources will be required to bolster the current capacity when undertaking project related work that does not fit within the current resource availability. This will need to be considered at the business case stage of the project to determine if the value created outweighs any associated costs.

When undertaking this type of procurement, the Council's Commercial function will be engaged to direct the use of appropriate frameworks to secure the resource while complying with procurement regulations.

Appropriate standards and requirements will be included in any procurement.

5. Risks

1) The main risk will be that the authority continues to develop services in a way that does not support digital ways of working. This will lead to the continuation of sub-optimal services that do not operate in a manner that supports integrated responses to our citizens.
   The mitigation is to introduce GDS Service Standards (ANNEX 2) to put in stage gates at each significant stage of a project.
2) The staff do not gain the skills to support the development of digital services.
   The mitigation will be to identify the staff training required as we embark on any new piece of work. Department for Leveling Up Housing and Communities (DLUPHC) will supply training to Local Authorities who have signed the Digital Declaration. Nottinghamshire has already signed the Digital Declaration and as such can leverage that training to give staff the training. Within our planning, time will need to be built in to enable staff to get up to speed as they become familiar with the techniques.
3) The priorities overwhelm the available resource to undertake digital work.
   Mitigation – A single Digital Board has been created to prioritise the Digital work that the Council undertakes. This will identify where the demands outstrips the Councils capacity and enable decisions to be made about if additional work should be funded. These decisions being made in light of the maximum capacity of the Council to sustain change.

6. Annexes

ANNEX 1

Agile Principles

What are the Agile Principles?

There are 12 agile principles outlined in The Agile Manifesto in addition to the 4 agile values. These 12 principles for agile software development help establish the tenets of the agile mindset. They are not a set of rules for practicing agile, but a handful of principles to help instill agile thinking.

Below we will review each of the 12 agile principles and describe how they may be practiced.

Agile Principle 1

“Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.”

The best ways to ensure you make customers happy while continuously delivering valuable software are to ship early, iterate frequently, and listen to your market continually.

Unlike traditional approaches to product development, which have notoriously long development cycles, agile principles encourage minimizing the time between ideation and launch. The idea is to get a working product in the hands of customers as soon as possible. Doing this successfully means product managers are able to quickly get a minimum viable product (MVP) out and into the world and use it to get feedback from real customers. This feedback is then fed back into the product development process and used to inform future releases.

How it looks in practice:

- Product teams use minimum viable products and rapid experimentation to test hypothesis and validate ideas.
- Frequent releases help fuel a continuous feedback cycle between customer and product.
- Shipped and done are not the same thing. Instead of releasing a “finished” product, iterations continue to make incremental improvements to product based on customer and market feedback.

Agile Principle 2

“Welcome changing requirements, even late in development. Agile processes harness change for the customer’s competitive advantage.”

In the world around us, change is the only constant. Agile principles and values support responding to these changes rather than moving forward in spite of them. Previous approaches to product development were often change adverse; detailed, well-documented plans were made before development began and were set in stone regardless of new findings. Agile principles support observing changing markets, customer needs, and competitive threats and changing course when necessary.

How it looks in practice:

- Product teams are guided by high-level strategic goals and perhaps even themes below those goals. The product department's success is measured against progress toward those strategic goals rather than by delivery of a predefined feature set.
- Product constantly has its ear to the ground monitoring the market, customer feedback, and other factors which could influence product direction. When actionable insight is uncovered, plans are adjusted to better serve customer and business needs.
- Product strategy and tactical plans are reviewed, adjusted, and shared on a regular cadence to reflect changes and new findings. As such, product needs to manage the expectations of executive stakeholders appropriately and ensure they understand the *why* behind changes.

Agile Principle 3

"Deliver working product frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale."

Agile philosophy favours breaking a product's development into smaller components and "shipping" those components frequently. Using an agile approach, therefore—and building in more frequent mini-releases of your product—can speed the product's overall development.

This agile approach, with short-term development cycles of smaller portions of the product, results in less time spent drafting and poring over the large amounts of documentation that characterizes Waterfall product development. More importantly, this frequent-release approach creates more opportunities for you and your teams to validate your product ideas and strategies from the qualified constituencies who see each new release.

How it looks in practice:

- Agile development cycles, often called "sprints" or "iterations" break down product initiatives into smaller chunks that can be completed in a set timeframe. Often this timeframe is between 2 and 4 weeks which truly is a sprint if you consider the marathon-like development cycles waterfall teams often follow.
- Another popular alternative to agile sprints is continuous deployment. This method of shipping software/product frequently works less in terms of predetermined time boxes and more in terms of simply deciding what to do and doing it.

Agile Principle 4

"Business people and developers must work together daily throughout the project."

Communication is a critical component of any project or team's success, and agile principles essentially mandate that it's a daily event. It takes a village to raise a child they say, and that applies to product as well.

A successful product requires insight from the business and technical sides of an organization which can only happen if these two teams work together consistently. Regular communication between business people and developers helps improve alignment across the organization by building trust and transparency.

How it looks in practice:

- Cross-functional agile product development teams include product people. This means that product is represented on the development team and bridges the gap between technical and business aspects of the product.
- Daily update meetings, or standups, are one technique many agile teams use to put this principle in practice and keep everyone connected.

Agile Principle 5

“Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.”

A key part of the agile philosophy is empowering individuals and teams through trust and autonomy. The agile team needs to be carefully built to include the right people and skill sets to get the job done, and responsibilities need to be clearly defined before the beginning of a project. Once the work has begun, however, there’s no place in agile for micromanagement or hand holding.

How it looks in practice:

- Product must clearly ensure engineering understands strategy and requirements before development starts. This means not only sharing user stories with the cross-functional team but also the bigger picture outlined in the product roadmap.
- Product is not responsible for explaining “how” something should be built. They need to share what and why, but it’s the delivery team’s job to determine the how. Furthermore, during sprints product does not micromanage outcome, instead they make themselves available to answer questions and provide support as needed.

Agile Principle 6

“The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.”

With so many distributed or remote development teams these days, this principle gets a bit of critique. But at the root of it, effective communication with developers means getting these conversations out of Slack and email and favoring more human interaction (even if done by video conference calls). The overall objective behind this principle is to encourage product people and developers to truly communicate in real time about the product, requirements, and the high-level strategy driving those things.

How it looks in practice:

- Daily standup meetings
- Collaborative backlog grooming sessions

- Sprint planning meetings
- Frequent demos
- Pair-programming

Agile Principle 7

“Working software is the primary measure of progress.”

Proponents of the agile philosophy are quick to remind us that we’re in the business of building software, and that’s where our time should be spent. Perfect, detailed documentation is secondary to working software. This mentality pushes to get products to the market quickly rather than let documentation or an “it’s not done until it’s perfect” mentality become a bottleneck. The ultimate measure for success is a working product that customers love.

How it looks in practice:

- Designing and releasing “Minimum Viable Features” rather than fully-developed feature sets means thinking first and foremost about the smallest things we can ship to start getting customer feedback and validate as we continue to build software.
- A fail fast mentality means moving forward even in times of uncertainty and testing ideas rapidly.
- Ship software often: a useful product now is better than a perfect one later.

Agile Principle 8

“Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.”

Keeping up with a demanding, rapid release schedule can be taxing on a team. Especially if expectations are set too high. Agile principles encourage us to be mindful of this and set realistic, clear expectations. The idea is to keep morale high and improve work-life balance to prevent burnout and turnover among members of cross functional teams.

How it looks in practice:

- Before every sprint, careful consideration of the amount of work that can be committed to is made. Development teams don’t over promise on what they can and cannot deliver. Effort estimations are a common practice in setting output expectations for development teams.
- Everyone agrees on what will get done during a sprint. Once a sprint has begun, no additional tasks are to be added except in rare cases.
- Product managers should act as gatekeepers to reduce the noise from other stakeholders and to avoid squeezing in additional unplanned work during an ongoing sprint.
- Product people should do their part in promoting a sense of psychological safety across the cross-functional team that encourages open communication and freely flowing feedback.

Agile Principle 9

“Continuous attention to technical excellence and good design enhances agility.”

While the agile philosophy encourages shorter cycles and frequent releases, it also puts emphasis on the importance of keeping things neat and tidy so they don’t cause problems in the future. Product managers often forget about this aspect of development because they mostly don’t spend their days wading through their products’ codebases, but it is still of the utmost importance to them.

How it looks in practice:

- The team needs to be cognizant of technical debt and the technical debt implications of any new features or initiatives added to the backlog. Developers and product need to work together to understand if and when technical debt is acceptable.
- On a regular basis, product will need to allocate development resources to refactoring efforts. Refactoring cannot be an afterthought, it needs to be an ongoing consideration.

Agile Principle 10

“Simplicity—the art of maximizing the amount of work not done—is essential.”

You’ve probably heard of the 80/20 rule—the concept that you can usually get 80% of your intended results with just 20% of the work. Agile principles encourage thinking this way; doing the things that can have the most impact. In a product management context this means having a laser sharp focus on organizational objectives and making some cutthroat prioritization decisions. Agile principles discourage building merely for the sake of building by emphasizing the importance of being strategic and building with purpose.

How it looks in practice:

- Product managers need to make very focused product decisions and closely align product strategy with organizational goals while being extremely picky about what user stories and features actually make the cut. Using prioritization techniques to prioritize initiatives by effort and predicted impact is one way product teams can apply this agile principle to product development.
- The short sprints that agile is characterized by present many opportunities for rapid testing and experimentation which can help reduce uncertainty around whether initiatives will truly have the predicted impact. Using experiments to validate ideas before building them up to spec is a great way to weed out bad ideas and identify good ones.

Agile Principle 11

“The best architectures, requirements, and designs emerge from self-organizing teams.”

In traditional software development methodologies, you’ll often see pyramid shaped teams where management makes key decisions for contributors. Agile principles suggest the use of self-organizing teams which work with a more “flat” management style where decisions are made as a group rather than by a singular manager or management team. The concept ties into agile’s value of teams and interactions over processes and tools, and the intent behind the concept is to empower teams to work together as they need to.

How it looks in practice:

- Self-organizing teams are autonomous groups within the organization who take control and responsibility over their respective projects and have ownership of those areas. Different organizations practice this principle differently. Spotify, for example uses “product squads” to practice this.

Learn more about managing complex requirements in an agile world in the webinar below.

Agile Principle 12

“At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.”

If you’re truly living by agile principles, there is no place for “we can’t change because we’ve always done it this way.” Just like we’re always learning new things about our customers and markets, we’re also learning from the processes we’re using to learn those things. Agile is not about following a strictly-defined process for every sprint and release, it’s about continuous improvement. And that continuous improvement must also extend to processes and teams.

How it looks in practice:

- Experimentation and testing is not limited to the product only. Agile teams are encouraged to experiment with their processes. You may think you’re already doing something well only to experiment with a revised version of the process and discover an even more effective method. Experimenting with your process and team is just as important as experimenting with the software you’re building.
- Regular retrospectives are opportunities for the team to discuss what went well, what didn’t go so well, and where the process can be tweaked to improve things in the future. They’re an excellent place for product managers and product owners to learn if they are communicating effectively with developers and giving them the support they need before, during, and after sprints.
- Another consideration to make regarding this agile principle is that in order to practice it effectively you need to create a culture of trust and transparency that encourages openness and frequent sharing of feedback.

ANNEX 2

This is a series of links to the Government Digital Services website that shows the service standards upon which any service should be assessed.

1. Understand users and their needs

Read more about point 1

2. Solve a whole problem for users

Read more about point 2

3. Provide a joined up experience across all channels

Read more about point 3

4. Make the service simple to use

Read more about point 4

5. Make sure everyone can use the service

Read more about point 5

6. Have a multidisciplinary team

Read more about point 6

7. Use agile ways of working

Read more about point 7

8. Iterate and improve frequently

Read more about point 8

9. Create a secure service which protects users’ privacy

Read more about point 9

10. Define what success looks like and publish performance data

Read more about point 10

11. Choose the right tools and technology

Read more about point 11

12. Make new source code open

Read more about point 12

13. Use and contribute to open standards, common components and patterns

Read more about point 13

14. Operate a reliable service

Read more about point 14

ANNEX 3

This document is designed to be used as a stand-alone document to inform the Design Authority of the technical architectural guiderails that the Authority has adopted.

ICT Architecture
Principles 2021-25 V

Glossary of Agile Terms

Backlog - A product backlog is a list of the new features, changes to existing features, bug fixes, infrastructure changes or other activities that a team may deliver in order to achieve a specific outcome.

The product backlog is the single authoritative source for things that a team works on. That means that nothing gets done that isn't on the product backlog. Conversely, the presence of a product backlog item on a product backlog does not guarantee that it will be delivered. It represents an option the team has for delivering a specific outcome rather than a commitment.

It should be cheap and fast to add a product backlog item to the product backlog, and it should be equally as easy to remove a product backlog item that does not result in direct progress to achieving the desired outcome or enable progress toward the outcome.

Burn downs/up - Burndown charts and burnup charts track the amount of output (in terms of hours, story points, or backlog items) a team has completed across an iteration or a project. (see more)