

Attachment 2: US Digital Service Playbook

The official US Digital Service Playbook can be located at: <https://playbook.cio.gov/>.

Summary of Proposed Project for Infiniti KMT prototype:

In this file Infiniti has added shaded boxes, like this one, to describe our prototype scenario, which is based around creating a KMT for a tax department, as well as answer the playbook questions (the ones that are appropriate for the prototype project).

Tax Department KMT Scenario:

As the Director of the Personal Income Tax Division, I want to increase the accuracy of personal income tax adjustments within our organization. I want to make available to our tax auditors information pertaining to the tax law, tax regulations, and our experts interpretation of these laws. In addition, I want to make available information on which tax adjustments or most profitable and likely to be upheld at the protest level, legal level, and within the courts.

Players:

Tax Auditors - Tax auditors review tax returns to identify possible tax adjustments that will result in additional income tax owed to the state. Tax auditors rely on their own case law research to make these adjustments. In some cases, they may be aware of certain positions the state has taken in regard to certain adjustments, but mostly they are making their own determinations based on their individual research.

Program Specialists (PS) - Program Specialists are responsible for interpreting the tax law and preparing positions for the state. They provide guidance and direction to Tax Auditors and considered the subject matter experts on tax law. Program Specialist's jobs are to provide guidance to Tax Auditors, however, this is on a case by case basis. In some cases they prepare reviews of the tax law, but these are hard to find.

Protest Hearing Officers (PHOs) – PHO's review the adjustments, perform their own research and either deny or approve the protest. PHO's collect information similar to Tax Auditors and make their own determination. There is no central place of knowledge on the state's position to make these determinations.

Play 1: Understand What People Need

We must begin digital projects by exploring and pinpointing the needs of the people who will use the service, and the ways the service will fit into their lives. Whether the users are members of the public or government employees, policy makers must include real people in their design process from the beginning. The needs of people — not constraints of government structures or silos — should inform technical and design decisions. We need to continually test the products we build with real people to keep us honest about what is important.

Checklist

- Early in the project, spend time with current and prospective users of the service
- Use a range of qualitative and quantitative research methods to determine people's goals, needs, and behaviors; be thoughtful about the time spent
- Test prototypes of solutions with real people, in the field if possible
- Document the findings about user goals, needs, behaviors, and preferences
- Share findings with the team and agency leadership
- Create a prioritized list of tasks the user is trying to accomplish, also known as "user stories"
- As the digital service is being built, regularly test it with potential users to ensure it meets people's needs

Key Questions

- Who are your primary users?
 - Tax Auditors
 - Program Specialists
 - Protest Hearing Officers
- What user needs will this service address?
 - The Knowledge Articles will help auditors understand how to correctly apply tax law which will lead to
 - Correct audit findings
 - Correct Notice of Proposed Change
 - Correct Taxable Income
- Why does the user want or need this service?
 - Our Tax Auditors need to understand the tax law, regulations, and our internal interpretation of the tax law, along with the potential of the adjustment being upheld at the protest, legal, and court level
- Which people will have the most difficulty with the service?
 - Program Specialist, will have the most interaction and will be creating most of the KAs. It will be their responsibility to update the KMT and approve certain direction moving forward on specific audit issues
- Which research methods were used?
 - Interviewed former Auditors, Program Specialist and Protest Hearing Officers
 - Research interviews with former auditor/PS/PHO gave details on how each currently completed their work assignments
 - Research interviews with former auditor/PS/PHO gave details on the the various tools and resources each used to complete their work
- What were the key findings?
 - Information is spread out across the organization
 - There are multiple interpretations of the tax law within even our own organization
 - The tax department is issuing notices at the audit level and protest is withdrawing the notices without informing the audit staff, so they continue making the same mistake
 - There is no central place to even find FAQ's, IRS Reg's or interpretations of the law
- How were the findings documented? Where can future team members access the documentation?
 - Multiple sessions, all information on the Confluence space
- How often are you testing with real people?
 - We did one round of usability testing with real people

Play 2: Address the Whole Experience, from Start to Finish

We need to understand the different ways people will interact with our services, including the actions they take online, through a mobile application, on a phone, or in person. Every encounter — whether it's online or offline — should move the user closer towards their goal.

Checklist

- Understand the different points at which people will interact with the service – both online and in person
- Identify pain points in the current way users interact with the service, and prioritize these according to user needs
- Design the digital parts of the service so that they are integrated with the offline touch points people use to interact with the service
- Develop metrics that will measure how well the service is meeting user needs at each step of the service

Key Questions

- What are the different ways (both online and offline) that people currently accomplish the task the digital service is designed to help with?

- Online: Ability to search for Knowledge Articles by subject matter
- Offline: Ability to meet in person and discuss the findings with subject matter experts, each document must have either an author or subject matter expert attached to it

- Where are user pain points in the current way people accomplish the task?

- Sporadic research, everyone starts from basic tax law, or the auditor training manual

- Where does this specific project fit into the larger way people currently obtain the service being offered?

- The project will allow for consistent direction on tax adjustments. The project would provide for a feedback mechanism when tax auditors are incorrect and cases are being withdrawn.
- Most importantly, the project will provide for a KMT for Program Specialist to publish their opinions on tax law and regulations, providing both Tax Auditors and PHO's the departments position.

- What metrics will best indicate how well the service is working for its users?

- How many adjustments are being withdrawn, before and after?
- Are the tax adjustments more consistent?
- Is the KMT being used?
- Are the P/S providing new findings in the KMT

Play 3: Make it Simple and Intuitive

Using a government service shouldn't be stressful, confusing, or daunting. It's our job to build services that are simple and intuitive enough that users succeed the first time, unaided.

Checklist

- Use a simple and flexible design style guide for the service. Use the [U.S. WebDesign Standards](#) as a default
- Use the design style guide consistently for related digital services
- Give users clear information about where they are in each step of the process
- Follow accessibility best practices to ensure all people can use the service
- Provide users with a way to exit and return later to complete the process
- Use language that is familiar to the user and easy to understand
- Use language and design consistently throughout the service, including online and offline touch points

Key Questions

- What primary tasks are the user trying to accomplish?
 - KMT that provides research and direction on the state's position regarding tax adjustments
- Is the language as plain and universal as possible?
 - Yes, the language is simple.
- What languages is your service offered in?
 - English
- If a user needs help while using the service, how do they go about getting it?
 - Help is available and integrated within KMT
- How does the service's design visually relate to other government services?
 - Visual design is simple and uses CSS and theme that can be adapted to any client

Play 4: Build the Service Using Agile and Iterative Practices

We should use an incremental, fast-paced style of software development to reduce the risk of failure. We want to get working software into users' hands as early as possible to give the design and development team opportunities to adjust based on user feedback about the service. A critical capability is being able to automatically test and deploy the service so that new features can be added often and be put into production easily.

Checklist

- Ship a functioning "minimum viable product" (MVP) that solves a core user need as soon as possible, no longer than three months from the beginning of the project, using a "beta" or "test" period if needed
- Run usability tests frequently to see how well the service works and identify improvements that should be made
- Ensure the individuals building the service communicate closely using techniques such as launch meetings, war rooms, daily standups, and team chat tools
- Keep delivery teams small and focused; limit organizational layers that separate these teams from the business owners
- Release features and improvements multiple times each month

- Create a prioritized list of features and bugs, also known as the “feature backlog” and “bug backlog”
- Use a source code version control system
- Give the entire project team access to the issue tracker and version control system
- Use code reviews to ensure quality

Key Questions

- How long did it take to ship the MVP? If it hasn’t shipped yet, when will it?
 • The Infiniti KMT MVP was finished in 10 days
- How long does it take for a production deployment?
 • The deployment is automated with Jenkins and completes in a few minutes
- How many days or weeks are in each iteration/sprint?
 • Sprints are two weeks.
- Which version control system is being used?
 • Github
- How are bugs tracked and tickets issued? What tool is used?
 • JIRA
- How is the feature backlog managed? What tool is used?
 • JIRA
- How often do you review and reprioritize the feature and bug backlog?
 • With each Sprint
- How do you collect user feedback during development? How is that feedback used to improve the service?
 • Usability testing notes from sessions.
- At each stage of usability testing, which gaps were identified in addressing user needs?
 • Minor gaps were identified during initial usability testing

Play 5: Structure Budgets and Contracts to Support Delivery

To improve our chances of success when contracting out development work, we need to work with experienced budgeting and contracting officers. In cases where we use third parties to help build a service, a well-defined contract can facilitate good development practices like conducting a research and prototyping phase, refining product requirements as the service is built, evaluating open source alternatives, ensuring frequent delivery milestones, and allowing the flexibility to purchase cloud computing resources.

[The TechFAR Handbook](#) provides a detailed explanation of the flexibilities in the Federal Acquisition Regulation (FAR) that can help agencies implement this play.

Checklist

- Budget includes research, discovery, and prototyping activities
- Contract is structured to request frequent deliverables, not multi-month milestones
- Contract is structured to hold vendors accountable to deliverables
- Contract gives the government delivery team enough flexibility to adjust feature prioritization and delivery schedule as the project evolves
- Contract ensures open source solutions are evaluated when technology choices are made
- Contract specifies that software and data generated by third parties remains under our control, and can be reused and released to the public as appropriate and in accordance with the law
- Contract allows us to use tools, services, and hosting from vendors with a variety of pricing

models, including fixed fees and variable models like “pay-for-what-you-use” services

- Contract specifies a warranty period where defects uncovered by the public are addressed by the vendor at no additional cost to the government
- Contract includes a transition of services period and transition-out plan

Key Questions

- What is the scope of the project? What are the key deliverables?

- KMT providing knowledge about tax adjustments
- Pilot project with Charitable Contributions Knowledge Articles

- What are the milestones? How frequent are they?

- Weekly sprints.

- What are the performance metrics defined in the contract (e.g., response time, system uptime, time period to address priority issues)?

Play 6: Assign One Leader and Hold that Person Accountable

There must be a single product owner who has the authority and responsibility to assign tasks and work elements; make business, product, and technical decisions; and be accountable for the success or failure of the overall service. This product owner is ultimately responsible for how well the service meets needs of its users, which is how a service should be evaluated. The product owner is responsible for ensuring that features are built and managing the feature and bug backlogs.

Checklist

- A product owner has been identified
- All stakeholders agree that the product owner has the authority to assign tasks and make decisions about features and technical implementation details
- The product owner has a product management background with technical experience to assess alternatives and weigh tradeoffs
- The product owner has a work plan that includes budget estimates and identifies funding sources
- The product owner has a strong relationship with the contracting officer

Key Questions

- Who is the product owner?

- Tax Department Director
- What organizational changes have been made to ensure the product owner has sufficient authority over and support for the project?
- What does it take for the product owner to add or remove a feature from the service?

- Tax Department Director would add to backlog in Jira

Play 7: Bring in Experienced Teams

We need talented people working in government who have experience creating modern digital services. This includes bringing in seasoned product managers, engineers, and designers. When outside help is needed, our teams should work with contracting officers who understand how to evaluate third-party technical competency so our teams can be paired with contractors who are good at both building and delivering effective digital services. The makeup and experience requirements of the team will vary depending on the scope of the project.

Checklist

- Member(s) of the team have experience building popular, high-traffic digital services

- The members of our team have experience building KMT for Educational organizations and many other digital services for other public and private sector organizations.
- Member(s) of the team have experience designing mobile and web applications

- Members of our team have designed mobile and web applications, including KMT solutions for public sector organizations.
- Member(s) of the team have experience using automated testing frameworks

- Members of our team have experience with automated testing frameworks including SonarQube, junit, nUnit, Selenium.
- Member(s) of the team have experience with modern development and operations (DevOps) techniques like continuous integration and continuous deployment

- Infiniti is an organization that specializes in DevOps. We have a five year contract to provide DevOps to CCCTC the largest higher-ed system in the world. CCCTC builds custom applications that are used by 114 colleges and 4 million students – Infiniti performs DevOps for all the CCCTC apps.
- Member(s) of the team have experience securing digital services

- Infiniti is an organization with extensive digital services security experience including

AWS and Microsoft Azure. Infiniti is experienced at performing NIST assessments.

- A Federal contracting officer is on the internal team if a third party will be used for development work
- A Federal budget officer is on the internal team or is a partner
- The appropriate privacy, civil liberties, and/or legal advisor for the department or agency is a partner

Play 8: Choose a Modern Technology Stack

The technology decisions we make need to enable development teams to work efficiently and enable services to scale easily and cost-effectively. Our choices for hosting infrastructure, databases, software frameworks, programming languages and the rest of the technology stack should seek to avoid vendor lock-in and match what successful modern consumer and enterprise software companies would choose today. In particular, digital services teams should consider using open source, cloud-based, and commodity solutions across the technology stack, because of their widespread adoption and support by successful consumer and enterprise technology companies in the private sector.

Checklist

- Choose software frameworks that are commonly used by private-sector companies creating similar services
- Whenever possible, ensure that software can be deployed on a variety of commodity hardware types
- Ensure that each project has clear, understandable instructions for setting up a local development environment, and that team members can be quickly added or removed from projects
- [Consider open source software solutions](#) at every layer of the stack

Key Questions

- What is your development stack and why did you choose it?

- Infiniti's team is experienced in the implementation of the Equalla KMT and has chosen it for this project because it is an open source framework that provides the necessary functionality for a KMT. Additional customization has been implemented in java and javascript.
- Which databases are you using and why did you choose them?

- Infiniti chose Postgres because it is an open source database.
- How long does it take for a new team member to start developing?

- Infiniti's developers start development within an hour because we use automation scripts to create development environments

Play 9: Deploy in a Flexible Hosting Environment

Our services should be deployed on flexible infrastructure, where resources can be provisioned in real-time to meet spikes in traffic and user demand. Our digital services are crippled when we host them in data centers that market themselves as “cloud hosting” but require us to manage and maintain hardware directly. This outdated practice wastes time, weakens our disaster recovery plans, and results in significantly higher costs.

Checklist

- Resources are provisioned on demand
- Resources scale based on real-time user demand
- Resources are provisioned through an API
- Resources are available in multiple regions
- We only pay for resources we use
- Static assets are served through a content delivery network
- Application is hosted on commodity hardware

Key Questions

- Where is your service hosted?

- AWS
- What hardware does your service use to run?

- AWS EC2 and RDS
- What is the demand or usage pattern for your service?
- What happens to your service when it experiences a surge in traffic or load?

- The AWS EC2 and ECS are setup to auto scale to meet surges
- How much capacity is available in your hosting environment?
- How long does it take you to provision a new resource, like an application server?
- How have you designed your service to scale based on demand?

- How are you paying for your hosting infrastructure (e.g., by the minute, hourly, daily, monthly, fixed)?
- Is your service hosted in multiple regions, availability zones, or data centers?
- In the event of a catastrophic disaster to a datacenter, how long will it take to have the service operational?
- What would be the impact of a prolonged downtime window?
- What data redundancy do you have built into the system, and what would be the impact of a catastrophic data loss?
- How often do you need to contact a person from your hosting provider to get resources or to fix an issue?

Play 10: Automate Testing and Deployments

Today, developers write automated scripts that can verify thousands of scenarios in minutes and then deploy updated code into production environments multiple times a day. They use automated performance tests which simulate surges in traffic to identify performance bottlenecks. While manual tests and quality assurance are still necessary, automated tests provide consistent and reliable protection against unintentional regressions, and make it possible for developers to confidently release frequent updates to the service.

Checklist

- Create automated tests that verify all user-facing functionality
- Create unit and integration tests to verify modules and components
- Run tests automatically as part of the build process
- Perform deployments automatically with deployment scripts, continuous delivery services, or similar techniques
- Conduct load and performance tests at regular intervals, including before public launch

Key Questions

- What percentage of the code base is covered by automated tests?
- How long does it take to build, test, and deploy a typical bug fix?
- How long does it take to build, test, and deploy a new feature into production?
- How frequently are builds created?
- What test tools are used?
- Which deployment automation or continuous integration tools are used?
- What is the estimated maximum number of concurrent users who will want to use the system?
- How many simultaneous users could the system handle, according to the most recent capacity test?
- How does the service perform when you exceed the expected target usage volume? Does it degrade gracefully or catastrophically?
- What is your scaling strategy when demand increases suddenly?

Play 11: Manage Security and Privacy through Reusable Processes

Our digital services have to protect sensitive information and keep systems secure. This is typically a process of continuous review and improvement which should be built into the development and maintenance of the service. At the start of designing a new service or feature, the team lead should engage the appropriate privacy, security, and legal officer(s) to discuss the type of information collected, how it should be secured, how long it is kept, and how it may be used and shared. The sustained engagement of a privacy specialist helps ensure that personal data is properly managed. In addition, a key process to building a secure service is comprehensively testing and certifying the components in each layer of the technology stack for security vulnerabilities, and then to re-use these same pre-certified components for multiple services.

The following checklist provides a starting point, but teams should work closely with their privacy specialist and security engineer to meet the needs of the specific service.

Checklist

- Contact the appropriate privacy or legal officer of the department or agency to determine whether a System of Records Notice (SORN), Privacy Impact Assessment, or other review should be conducted
- Determine, in consultation with a records officer, what data is collected and why, how it is used or shared, how it is stored and secured, and how long it is kept
- Determine, in consultation with a privacy specialist, whether and how users are notified about how personal information is collected and used, including whether a privacy policy is needed and where it should appear, and how users will be notified in the event of a security breach
- Consider whether the user should be able to access, delete, or remove their information from the service
- “Pre-certify” the hosting infrastructure used for the project using FedRAMP
- Use deployment scripts to ensure configuration of production environment remains consistent and controllable

Key Questions

- Does the service collect personal information from the user?
How is the user notified of this collection?
- Does it collect more information than necessary? Could the data be used in ways an average user wouldn't expect?
- How does a user access, correct, delete, or remove personal information?
- Will any of the personal information stored in the system be shared with other services, people, or partners?
- How and how often is the service tested for security vulnerabilities?
- How can someone from the public report a security issue?

Play 12: Use Data to Drive Decisions

At every stage of a project, we should measure how well our service is working for our users. This includes measuring how well a system performs and how people are interacting with it in real-time. Our teams and agency leadership should carefully watch these metrics to find issues and identify which bug fixes and improvements should be prioritized. Along with monitoring tools, a feedback mechanism should be in place for people to report issues directly.

Checklist

- Monitor system-level resource utilization in real time
- Monitor system performance in real-time (e.g. response time, latency, throughput, and error rates)
- Ensure monitoring can measure median, 95th percentile, and 98th percentile performance
- Create automated alerts based on this monitoring
- Track concurrent users in real-time, and monitor user behaviors in the aggregate to determine how well the service meets user needs
- Publish metrics internally
- Publish metrics externally
- Use an experimentation tool that supports multivariate testing in production

Key Questions

- What are the key metrics for the service?
 - Usage and viewing of Knowledge Articles
- How have these metrics performed over the life of the service?
- Which system monitoring tools are in place?
 - NewRelic and AWS CloudWatch
- What is the targeted average response time for your service?
What percent of requests take more than 1 second, 2 seconds, 4 seconds, and 8 seconds?
- What is the average response time and percentile breakdown (percent of requests taking more than 1s, 2s, 4s, and 8s) for the top 10 transactions?
- What is the volume of each of your service's top 10 transactions? What is the percentage of transactions started vs. completed? What is your service's monthly uptime target?
- What is your service's monthly uptime percentage, including scheduled maintenance? Excluding scheduled maintenance?
- How does your team receive automated alerts when incidents occur?
- How does your team respond to incidents? What is your post-mortem process?
- Which tools are in place to measure user behavior?
- What tools or technologies are used for A/B testing?
- How do you measure customer satisfaction?

Play 13: Default to open

When we collaborate in the open and publish our data publicly, we can improve Government together. By building services more openly and publishing open data, we simplify the public's access to government services and information, allow the public to contribute easily, and enable reuse by entrepreneurs, nonprofits, other agencies, and the public.

Checklist

- Offer users a mechanism to report bugs and issues, and be responsive to these reports
- Provide datasets to the public, in their entirety, through bulk downloads and APIs (application programming interfaces)
- Ensure that data from the service is explicitly in the public domain, and that rights are waived globally via an international public domain dedication, such as the "Creative Commons Zero" waiver
- Catalog data in the agency's enterprise data inventory and add any public datasets to the agency's public data listing
- Ensure that we maintain the rights to all data developed by third parties in a manner that is releasable and reusable at no cost to the public
- Ensure that we maintain contractual rights to all custom software developed by third parties in a manner that is publishable and reusable at no cost
- When appropriate, create an API for third parties and internal users to interact with the service directly

- When appropriate, publish source code of projects or components online
- When appropriate, share your development process and progress publicly

Key Questions

- How are you collecting user feedback for bugs and issues?

- JIRA

- If there is an API, what capabilities does it provide? Who uses it? How is it documented?

- Yes there is an API and it is documented in Swagger

- If the codebase has not been released under an open source license, explain why.

- The code base is entirely Open Source

- What components are made available to the public as open source?
- What datasets are made available to the public?