

Digital Services Playbook

Best Practices for Building Digital Assets

<https://playbook.cio.gov/>

Play 1 - Understand what people need

- ☒ Early in the project, spend time with current and prospective users of the service
Conducted Initial User Interviews to design user personas
- ☐ 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
Performed usability testing and documented feedback as task
- ☒ Document the findings about user goals, needs, behaviors, and preferences
Placed findings in backlog as task
- ☒ Share findings with the team and agency leadership
User Interviews are documented and demoed at regular interviews with stakeholders just like new features
- ☒ Create a prioritized list of tasks the user is trying to accomplish, also known as "user stories"
JIRA KANBAN Board
- ☒ As the digital service is being built, regularly test it with potential users to ensure it meets people's needs
Regular User Demo's are presented to the team and stakeholders

Play 2 - Address the whole experience, from start to finish

- ☒ Understand the different points at which people will interact with the service – both online and in person
Created User Personas
- ☐ 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
We used Google Analytics to gather metrics.

Play 3 - Make it simple and intuitive

- ☒ Create or use an existing, simple, and flexible design style guide for the service
Design Style Guide
- ☒ Use the design style guide consistently for related digital services
Colors, fonts and guides were used for UI development
- ☐ 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

Play 4 - Build the service using agile and iterative practices

- ☐ 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
HipChat was used by the entire team and integrated into our build environment providing notifications when code was checked in, status of the build, new task and code reviews
- ☒ Keep delivery teams small and focused; limit organizational layers that separate these teams from the business owners
Development team consisted of 7 people, 4 developers, a product owner, designer and a UI/UX expert.
- ☐ 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”
JIRA KANBAN Board
- ☒ Use a source code version control system
GitHub
- ☒ Give the entire project team access to the issue tracker and version control system
Entire team has access to entire development and CI environment
- ☒ Use code reviews to ensure quality
Team used Crucible for Distributed Team Reviews, all assets and code were reviewed

Play 5 - Structure budgets and contracts to support delivery

- ☐ 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

Play 6 - Assign one leader and hold that person accountable

- ☒ A product owner has been identified
Chad Dalton as Product Owner
- ☒ All stakeholders agree that the product owner has the authority to assign tasks and make decisions about features and technical implementation details
Chad Dalton was given full authority to prioritize the backlog, task the team and speak on behalf of the customer for time critical decisions related to features and implementation
- ☒ The product owner has a product management background with technical experience to assess alternatives and weigh tradeoffs
Chad Dalton has successfully ran multiple agile projects across multiple teams
- ☒ The product owner has a work plan that includes budget estimates and identifies funding sources
Resources allocation was made at the beginning of the project
- ☐ The product owner has a strong relationship with the contracting officer

Play 7 - Bring in experienced teams

- ☒ Member(s) of the team have experience building popular, high-traffic digital services|
The team consisted to experienced Full Stack Developers that have worked on multiple web related applications
- ☒ Member(s) of the team have experience designing mobile and web applications
The team has built many mobile and web based solutions

- ☒ Member(s) of the team have experience using automated testing frameworks
The team has used Test Driven Development for many years and has used mocha and selenium for the past several
- ☒ Member(s) of the team have experience with modern development and operations (DevOps) techniques like continuous integration and continuous deployment
The team consisted of a DevOps engineer that has extensive experience configuring Bamboo CI and Jenkins CI for continuous integration and deployment
- ☒ Member(s) of the team have experience securing digital services
The team consisted of a security engineer that and automated security checks that ensured secure application deployment
- ☐ 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

- ☒ Choose software frameworks that are commonly used by private-sector companies creating similar services
Bootstrap, Angular, Swagger, Nodejs, Leaflet
- ☒ Whenever possible, ensure that software can be deployed on a variety of commodity hardware types
Developers used a range of environments including OSX, Linux, Ubuntu and Windows
- ☒ 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
README.md files provides clear instructions and a Dockerfile is also included along with a pre-build docker image in docker hub
- ☒ Consider open source software solutions at every layer of the stack
Entire stack was MIT or BSD licensed

Play 9 - Deploy in a flexible hosting environment

- ☒ Resources are provisioned on demand
Autoscaling in Amazon Web Services
- ☒ Resources scale based on real-time user demand
Autoscaling in Amazon Web Services
- ☒ Resources are provisioned through an API
AWS Stack supports API Provisioning
- ☒ Resources are available in multiple regions
Beanstalk App in Multi-Regions
- ☒ We only pay for resources we use
AWS is Pay as you go pricing model
- ☒ Static assets are served through a content delivery network
Entire UI portion deployed in AWS Cloudfront CDN
- ☒ Application is hosted on commodity hardware
Application hosted in AWS Elastic Beanstalk on Cloud Hardware

Play 10 - Automate testing and deployments

- ☒ Create automated tests that verify all user-facing functionality
Mocha Test ran as part of build process
- ☒ Create unit and integration tests to verify modules and components
Selenium UI/Integration Test Client/Server API Functionality
- ☒ Run tests automatically as part of the build process
Bamboo
- ☒ Perform deployments automatically with deployment scripts, continuous delivery services, or similar techniques
Bamboo
- ☐ Conduct load and performance tests at regular intervals, including before public launch

Play 11 - Manage security and privacy through reusable processes

- ☐ 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
AWS is FedRAMP Certified
- ☒ Use deployment scripts to ensure configuration of production environment remains consistent and controllable
Gruntfiles used for system deployment by Bamboo CI Server

Play 12 - Use data to drive decisions

- ☒ Monitor system-level resource utilization in real time
CloudWatch Alarms
- ☒ Monitor system performance in real-time (e.g. response time, latency, throughput, and error rates)
CloudWatch Alarms
- ☒ Ensure monitoring can measure median, 95th percentile, and 98th percentile performance
CloudWatch Metric showing Average
- ☒ Create automated alerts based on this monitoring
CloudWatch provides alarms that can send alerts based on configurable thresholds
- ☐ 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

Play 13 - Default to open

- ☐ 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
API Created First as part of development
- ☒ When appropriate, publish source code of projects or components online
Published on GitHub
- ☒ When appropriate, share your development process and progress publicly
All code and process assets are posted in the GitHub repo.