

U.S. DIGITAL SERVICES PLAYBOOK – CHHS PROTOTYPE

INTRODUCTION

Natoma Technologies, Inc. (Natoma) created this template to provide a concrete mechanism to ensure that an application is measurably aligned with the U.S. Digital Services Playbook¹. The questions and checklists in the template are copied from the Playbook and reformatted to be edited with project specifics. *Ensure that the checklist items are represented in the project plan, and enter project specific answers directly below the questions using italics to distinguish the answers from the prompts.*

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

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

Key Questions

1. Who are your primary users?

Working with a Subject Matter Expert (see notes <ref>), we identified two primary users, and the potential third user type

- *Foster Parents – which we call Resource Parents*

¹ <https://playbook.cio.gov>

Request for Information – RFI #75001

- *Case Workers – representing the County, or other authority*

Potential third user type:

- *Social Worker*

2. What user needs will this service address?

Needs:

- *Foster Parents –*
 - *Create and maintain their profile data that is significant to their case worker including certifications, distance from schools, etc*
 - *Ability to directly and securely communicate with their foster kid's case worker, on a variety of subjects, including incident reporting and progress. Secure communication will allow transmission of sensitive, health, and personal identifying information that would not be permitted over unsecured channels.*
- *Case Workers – representing the County, or other authority*
 - *Ease of access to up-to-date information about resource parents including certifications, distance from schools*
 - *Ability to directly and securely communicate with foster kid's resource parents, on a variety of subjects, including incident reporting and progress. Secure communication will allow transmission of sensitive, health, and personal identifying information that would not be permitted over unsecured channels.*
 - *Access to detailed profile information on the foster child, including recent medical and dental visits. This is out-of-scope for the prototype and was put in the product backlog.*
- *Social Workers: Explored the value of the case worker being able to share updates and information about the foster child with the social worker. This is out-of-scope for the prototype and was put in the product backlog.*
- *Biological parents: Explored the value of the biological parent being able to request information on their child, or request a visit from the case worker or being able to receive information about their child from the case worker. However, based on our user interviews, it was determined that communication would be focused on the foster parent. This is out of scope for the prototype.*

3. Why does the user want or need this service?

1. *Convenience*
2. *Secure communication will allow transmission of sensitive, health, and personal identifying information that would not be permitted over unsecured channels.*

4. Which people will have the most difficulty with the service?

Resource Parents that are not sufficiently familiar with technology

Resource Parents that have literacy concerns

Request for Information – RFI #75001

Noted the need for the site to be multi-lingual. This led to an architecture decision to use techniques that simplify adding multi-lingual support.

5. Which research methods were used?

Interviewing a subject matter expert

6. What were the key findings?

The depth and quantity of information that needs to be tracked and communicated to improve the fostering experience is higher than initially anticipated.

7. How were the findings documented? Where can future team members access the documentation?

The team wrote the interview notes, derived findings, and added them to the repository for future use.

8. How often are you testing with real people?

For the prototype, with its one week sprints, we tested with real people at a minimum of each sprint.

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

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

Key Questions

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

Request for Information – RFI #75001

Currently, our interview with the subject matter expert revealed that, the communication of key pieces of information about both the Resource Parent and the Foster Kid, for example date of last medical checkup, is accomplished during in-person and telephone communication.

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

A key pain point identified by the subject matter expert interview is that the various parties (Resource Parent, Case Worker, and Social Worker) are making decisions based on information that is out of date, or not shared by all parties, or both.

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

A prototype such as this, will inform the bigger picture as to how digital services will be received and consumed by these individuals.

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

The frequency with which volatile data (foster kid's school performance for example) is updated by the Resource Parent, aka freshness. The frequency of use of the secure messaging feature and the degree to which it replaces phone calls/voice mail.

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

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

Key Questions

1. What primary tasks are the user trying to accomplish?

Creating and maintaining a data store containing key attributes of the Resource Parent.

Request for Information – RFI #75001

Securely communicating with a Case Worker

2. Is the language as plain and universal as possible?

Iteration Two, of the UI, includes having users interact directly with the web site and ensuring that the labels, banners, and other key items are clear, and updating them and re-experimenting if not.

3. What languages is your service offered in?

Prototype is in English. Subject Matter Expert identified that the next most important languages are Spanish, then Russian, then Hmong. This led to an architecture decision to use techniques that simplify adding multi-lingual support, but it is unlikely that the translation will be accomplished during this timeframe.

4. If a user needs help while using the service, how do they go about getting it?

For the prototype there is no live-help service being offered.

5. How does the service's design visually relate to other government services?

The prototype visual aspect has been kept neutral.

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

1. 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
2. Run usability tests frequently to see how well the service works and identify improvements that should be made
3. Ensure the individuals building the service communicate closely using techniques such as launch meetings, war rooms, daily standups, and team chat tools
4. Keep delivery teams small and focused; limit organizational layers that separate these teams from the business owners
5. Release features and improvements multiple times each month
6. Create a prioritized list of features and bugs, also known as the "feature backlog" and "bug backlog"

Request for Information – RFI #75001

7. Use a source code version control system
8. Give the entire project team access to the issue tracker and version control system
9. Use code reviews to ensure quality

Key Questions

1. How long did it take to ship the MVP? If it hasn't shipped yet, when will it?

MVP was deployed after the first sprint (one week).

2. How long does it take for a production deployment?

After the automation is established, this is anticipated to be a matter of minutes

3. How many days or weeks are in each iteration/sprint?

Each sprint takes one week. This is shorter than the team's regular cycle as the timeframe for the prototype is short.

4. Which version control system is being used?

Team Foundation Services (TFS) from Visual Team Services with final copy for prototype submittal in GitHub.com.

5. How are bugs tracked and tickets issued? What tool is used?

In addition to automated testing, the team wrote test cases to assert the correct functioning of the prototype. Defects/bugs identified are created in TFS, triaged, assigned priority and severity, and tracked to completion.

6. How is the feature backlog managed? What tool is used?

Features are written as stories, which are entered into TFS as User Stories. During Backlog Grooming the Product Owner prioritized the stories, and during Sprint Planning, the team moved the Stories that were committed to into the Sprint Backlog and kept the remaining in the Product Backlog.

7. How often do you review and reprioritize the feature and bug backlog?

At the end of each Sprint – so, once a week for the prototype.

8. How do you collect user feedback during development? How is that feedback used to improve the service?

In addition to scheduling reviews with subject matter experts that approximate a user, the prototype is continuously available to the entire team, and all team members spend time trying it out, and

Request for Information – RFI #75001

seeing the progress being made. Team members make improvement suggestions by writing user stories that simply include the minimum (as a role, I want to X so that value) allowing us to capture ideas, improvements, and suggestions.

9. At each stage of usability testing, which gaps were identified in addressing user needs?

Providing a mechanism for forgot password as well as a mechanism for performing proximity searches. These items were added to the backlog, but it was identified that having these included would improve usability.

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

1. Budget includes research, discovery, and prototyping activities
2. Contract is structured to request frequent deliverables, not multi-month milestones
3. Contract is structured to hold vendors accountable to deliverables
4. Contract gives the government delivery team enough flexibility to adjust feature prioritization and delivery schedule as the project evolves
5. Contract ensures open source solutions are evaluated when technology choices are made
6. 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
7. 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
8. Contract specifies a warranty period where defects uncovered by the public are addressed by the vendor at no additional cost to the government
9. Contract includes a transition of services period and transition-out plan

Key Questions

1. What is the scope of the project? What are the key deliverables?

Request for Information – RFI #75001

Prototype scope is to demonstrate Natoma's ability to, using Agile techniques, build a prototype web site that demonstrates our processes for human-interface design, coupled with technical abilities in design, and building, using modern dev-ops techniques.

2. What are the milestones? How frequent are they?

Key milestone for the prototype were the release of the RFI on 5/5/2016 and the due date of 6/9/2016.

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

None for this prototype

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

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

Key Questions

1. Who is the product owner?

Veronica Westlund, Vice President & COO of Natoma Technologies, Inc.

2. What organizational changes have been made to ensure the product owner has sufficient authority over and support for the project?

There were none required for the development of this prototype.

3. What does it take for the product owner to add or remove a feature from the service?

Request for Information – RFI #75001

The Product Owner would prioritize one or more User Stories (or an epic) corresponding to the feature change or addition required. The team would then, during Sprint Planning, select stories that, with the highest priority, they can commit to delivering during the sprint.

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

1. Member(s) of the team have experience building popular, high-traffic digital services
2. Member(s) of the team have experience designing mobile and web applications
3. Member(s) of the team have experience using automated testing frameworks
4. Member(s) of the team have experience with modern development and operations (DevOps) techniques like continuous integration and continuous deployment
5. Member(s) of the team have experience securing digital services
6. A Federal contracting officer is on the internal team if a third party will be used for development work
7. A Federal budget officer is on the internal team or is a partner
8. 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

1. Choose software frameworks that are commonly used by private-sector companies creating similar services

Request for Information – RFI #75001

2. Whenever possible, ensure that software can be deployed on a variety of commodity hardware types
3. 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
4. Consider open source software solutions at every layer of the stack

Key Questions

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

Natoma is technology-neutral, and we want this prototype to demonstrate the importance of User Centric Design to deliver life improvements to citizens, and to showcase that Agile teams can nimbly build value that is supported by right-sized documentation, models, and best practices using a repeatable process.

We know from reading the RFI that the client wants to see modern, open, choices. We also have to take into account available resources and our time to market.

A popular choice for State of California is Microsoft's ASP.NET and related technologies, and Natoma recognizes that California is deeply vested in the ASP.NET/IIS stack, through software projects, tool and infrastructure acquisition, and staff training. Microsoft offers open version of its Visual Studio tools and platform that can be used at no cost for this prototype.

During the Q&A Web Session the RFI team confirmed that the entire solution need not be open source, but that at least 5 modern open source tools must be used. This confirmation solidified our technology choice.

This informed our decision to use ASP.net/IIS/Windows.

Development tools and libraries include:

Tool	Usage	License
<i>Visual Studio Community 2015</i>	<i>Integrated development environment for ASP.NET</i>	<i>Microsoft, refer to https://www.visualstudio.com/en-us/dn877550.aspx</i>
<i>MVC 4</i>	<i>UI Framework</i>	<i>Apache License 2.0</i>
<i>AutoMapper</i>	<i>Class Mapping</i>	<i>MIT</i>

Request for Information – RFI #75001

<i>Bootstrap</i>	<i>UI Layout</i>	<i>MIT</i>
<i>Moq</i>	<i>Unit Test Support</i>	<i>BSD 2- Clause</i>
<i>Knockout</i>	<i>Javascript Framework</i>	<i>MIT</i>
<i>jQuery</i>	<i>Javascript Framework</i>	<i>MIT</i>
<i>Visual Team Services/TFS</i>	<i>Source Control/Continuous Integration</i>	<i>Proprietary but available free basic license</i>
<i>Wave</i>	<i>Web Accessibility Evaluation Tool</i>	<i>MIT?</i>
<i>ChromeVox</i>	<i>Web Accessibility Evaluation Tool</i>	<i>Apache License 2.0</i>
<i>Application Insights</i>	<i>Continuous Monitoring</i>	<i>Proprietary</i>
<i>Google Analytics</i>	<i>Usage Tracking</i>	<i>Google, refer to https://www.google.com/analytics/terms/us.html</i>

2. Which databases are you using and why did you choose them?

For the prototype we did not choose a standard database, our data constructs were simple, the number of users expected are low, and so we persisted the data in XML files on the application server.

3. How long does it take for a new team member to start developing?

Our use of cloud technologies, specifically Visual Studio Team Services has shortened our startup time for developers to a matter of minutes. Requiring the addition of an account, and assignment of privileges to the applicable repositories.

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 traffic and user demand. Our digital services are crippled when we host

Request for Information – RFI #75001

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

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

Key Questions

1. Where is your service hosted?

We are hosting the prototype on Amazon Web Services.

2. What hardware does your service use to run?

The hardware is abstracted by the cloud services used and that information is not provided by the service. The service runs on a virtual machine.

3. What is the demand or usage pattern for your service?

Amazon does not publish the demand or usage patterns for their offering. However, the prototype is not expected to have high demand, and usage is expected mostly during the evaluation period.

4. What happens to your service when it experiences a surge in traffic or load?

Amazon’s cloud services are very robust and we do not anticipate that there will be any issues with traffic or load. However, if we did expect that we would set up elastic load balancing on AWS for our services.

5. How much capacity is available in your hosting environment?

Amazon’s cloud services have the ability to scale their capacity to within any capacity demands that we could foresee for this prototype.

6. How long does it take you to provision a new resource, like an application server?

It takes about 5 minutes to provision a new Windows application server.

7. How have you designed your service to scale based on demand?

Request for Information – RFI #75001

Based on the expected use we have not designed it to scale at this time. However, we would design it to use AWS Elastic Load Balancing and/or Auto Scaling if we expected a large demand.

8. How are you paying for your hosting infrastructure (e.g., by the minute, hourly, daily, monthly, fixed)?

Currently it is by the minute, but AWS offers a fixed cost if reserving for a minimum of one year.

9. Is your service hosted in multiple regions, availability zones, or data centers?
At this time it is not, it is only in one region. It does not have redundancy in other regions.
10. In the event of a catastrophic disaster to a datacenter, how long will it take to have the service operational?

Given that we have a public AMI for a prototype, the data does not need to be maintained, as such it would take 10 minutes to have the service operational, though the DNS proliferation may take up to 2 hours or so.

11. What would be the impact of a prolonged downtime window?

The objective for the prototype is to allow the State to evaluate a sample website built by Natoma. If there were a prolonged downtime that would prevent this objective. If there were a prolonged downtime window during the evaluation, the impact would be severe. However, we have a continuous monitoring tool that will alert us when the site is not available. Having this notification will allow us to review the issue and determine if a new instance is necessary to recover.

12. What data redundancy do you have built into the system, and what would be the impact of a catastrophic data loss?

For the purposes of this prototype data redundancy was not within scope.

13. How often do you need to contact a person from your hosting provider to get resources or to fix an issue?

We have not had to get resources or to fix an issue as yet.

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.

Request for Information – RFI #75001

Checklist

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

Key Questions

1. What percentage of the code base is covered by automated tests?

Natoma estimates that 75% of the code is covered by automated testing, but the toolset used does not provide a percentage coverage.

2. How long does it take to build, test, and deploy a typical bug fix?

Less than four hours.

3. How long does it take to build, test, and deploy a new feature into production?

New features have not yet been added to the prototype.

4. How frequently are builds created?

The source code is automatically built on each checked in change to the code. In addition to compilation, unit tests are automatically executed and the results made available to the team.

5. What test tools are used?

The following test tools are used:

- *Unit testing / Unit integration testing are supported by Moq*
- *Unit testing / Unit integration test execution are automated by Microsoft Visual Studio Team Services*
- *WebAim's WAVE, and Chrome VOX for accessibility evaluation*

6. Which deployment automation or continuous integration tools are used?

Microsoft Visual Studio Team Services provides continuous integration for build and releases. The project is currently being configured to automated deployment to a dev/test environment.

Request for Information – RFI #75001

7. What is the estimated maximum number of concurrent users who will want to use the system?

This prototype is anticipated to have ten or fewer concurrent users.

8. How many simultaneous users could the system handle, according to the most recent capacity test?

This prototype did not include formal capacity testing. Informal testing demonstrated that five concurrent users resulted in no performance degradation.

9. How does the service perform when you exceed the expected target usage volume? Does it degrade gracefully or catastrophically?

This prototype did not include formal capacity testing. Informal testing demonstrated that five concurrent users resulted in no performance degradation.

10. What is your scaling strategy when demand increases suddenly?

This prototype did not include a scaling strategy do to its limited nature.

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

1. 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
2. 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
3. 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

Request for Information – RFI #75001

is needed and where it should appear, and how users will be notified in the event of a security breach

4. Consider whether the user should be able to access, delete, or remove their information from the service
5. "Pre-certify" the hosting infrastructure used for the project using FedRAMP
6. Use deployment scripts to ensure configuration of production environment remains consistent and controllable

Key Questions

1. Does the service collect personal information from the user? How is the user notified of this collection?

The prototype requests a range of information, some of it personal and sensitive, from the Parent Resource user. The user voluntarily supplies it to their degree of comfort. Care was taken during prototype development to indicate to users whether or not the site area is sufficiently secure for the storage and transmission of health, minor, and other personal identifying information.

2. Does it collect more information than necessary? Could the data be used in ways an average user wouldn't expect?

For the purpose of designing and building a prototype, the set of information to request from the users was guided by our subject matter expert. For a real system, we would work with the State to ensure the appropriateness of data collected and used.

3. How does a user access, correct, delete, or remove personal information?

The user has control over all information stored by the prototype, and can log-in to access and edit their information.

4. Will any of the personal information stored in the system be shared with other services, people, or partners?

No.

5. How and how often is the service tested for security vulnerabilities?

The prototype does not contain sensitive data, if it did, operational procedures would have to be in place to detect and respond to security vulnerabilities.

6. How can someone from the public report a security issue?

End-users have the ability to provide feedback directly through a link on the About pages and on the Error page.

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

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

Key Questions

1. What are the key metrics for the service?

There are none for this prototype

2. How have these metrics performed over the life of the service?

Not applicable for this prototype

3. Which system monitoring tools are in place?

Natoma has deployed Application Insights to continuously monitor the availability of the prototype via 'ping' results. Were this a high availability, high criticality, application Natoma would recommend monitoring the vitals (processor, memory, disk space) of the servers in each tier in addition to monitoring the network health. However, server and AWS alerts can be added for this level of monitoring as well.

4. 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?

The prototype has not formally targeted an average response time.

Request for Information – RFI #75001

5. 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?

Response time testing has not been performed for this prototype.

6. What is the volume of each of your service's top 10 transactions? What is the percentage of transactions started vs. completed?

This is not yet known for this prototype.

7. What is your service's monthly uptime target?

Not applicable for this prototype

8. What is your service's monthly uptime percentage, including scheduled maintenance? Excluding scheduled maintenance?

This is not yet known for this prototype.

9. How does your team receive automated alerts when incidents occur?

Natoma has deployed Application Insights to continuously monitor the availability of the prototype, outage alerts are received by automated emails.

10. How does your team respond to incidents? What is your post-mortem process?

Due to the limited exposure of the prototype, there have been no incidents; however, Natoma has deployed Application Insights to continuously monitor the availability of the prototype. Were the prototype to contain sensitive data, operational procedures would have to be in place to detect and react to security incidents.

11. Which tools are in place to measure user behavior?

Natoma has integrated Google Analytics into the prototype.

12. What tools or technologies are used for A/B testing?

A/B testing has not been performed, Natoma has integrated Google Analytics into the prototype and this would support gathering analytics during A/B testing.

13. How do you measure customer satisfaction?

This is outside the scope of this prototype.

PLAY 13 DEFAULT TO OPEN

Request for Information – RFI #75001

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

1. Offer users a mechanism to report bugs and issues, and be responsive to these reports
2. Provide datasets to the public, in their entirety, through bulk downloads and APIs (application programming interfaces)
3. 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
4. Catalog data in the agency's enterprise data inventory and add any public datasets to the agency's public data listing
5. 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
6. 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
7. When appropriate, create an API for third parties and internal users to interact with the service directly
8. When appropriate, publish source code of projects or components online
9. When appropriate, share your development process and progress publicly

Key Questions

1. How are you collecting user feedback for bugs and issues?

End-users have the ability to provide feedback directly through a link on the About pages and on the Error page.

In addition, the site records page-by-page activity via Google Analytics and analysis of its metrics may provide insight into difficulties experienced by end-users.

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

For this prototype, there is no Application Programming Interface

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

The source code has been published to a public GitHub.com repository.

4. What components are made available to the public as open source?

Request for Information – RFI #75001

The limited prototype scope did not lend itself to reusable, original, components.

5. What datasets are made available to the public?

The prototype makes use of a CHHS publically available dataset, but the prototype itself has created no new datasets.