FAS2-UI-Stateless vs FAS2-UI

Framework differences

	FAS2-UI	FAS2-UI-Stateless
Deployment	Tomcat server with WAR deployables	Spring Boot executable JAR as deployable
Frontend engine	Thymeleaf	React JS
Backend framework	Spring (4.3.22)	Spring Boot (2.2.5)

Functional differences

The new FAS2-UI implementation has no functional differences and behaves the same as the old implementation.

Technical differences

FAS2-UI-Stateless provides more stability throughout the authentication process and is less error prone.

The complete overhaul of the project allows developers to maintain this application more easily. The code has a logical structure which is easy to understand, unlike the old version. More understandable code leads to a lot of great benefits.

The application is structured in a way that makes the introduction of new features very easy. The React Framework works with components for each feature.

The complete separation between frontend and backend allows for a smooth development process. Frontend changes are loaded immediately. This separation also allows the backend to be monitored more easily since the backend will not return HTML but will simply return data based on the endpoint you call.

FAS2-UI-Stateless also introduces a complete overhaul of the logging system. Every incoming request will include a correlation-ID. This ID will also be shown to the end-user in case of an unexpected error. This makes the debugging and problem-solving process of incoming incidents a breeze compared to the old logging system, since the old application does not guarantee a correlation-ID or does not guarantee logging at all in certain services.

Stateful vs Stateless

Stateful applications store client session data in the application while a Stateless application does not store any client information on the server. Since the FAS2-UI works in correlation with FAS, the FAS2-UI should not store the authentication information again, since this information is already stored in a FAS session.

Not storing this authentication information in the FAS2-UI application leads to less concurrency between FAS and FAS2-UI and guarantees that the correct data is used in the FAS2-UI authentication flow.

Deployment differences

Legacy FAS2-UI is run on a Tomcat server. This server will start up .war files in its instance. While the new FAS2-UI is a simple executable JAR which includes an embedded Tomcat. This makes FAS2-UI-Stateless easier to deploy. You simply execute the JAR and everything is done. No need for external Tomcat servers or extra software. Everything is embedded in the .JAR. You simply need the correct Java version on the server.

In short

	FAS2-UI	**FAS2-UI-Stateless**
Deployment	Tomcat server in combination with separate .WAR files	Executable JAR
Frontend architecture	Stateful	Stateless
Code quality	Outdated & error prone code structure	Easy to understand & high in maintainability
Expandability	Poor	Good
Logging	No CorrelationID & poor logging	CorrelationID & complete logging
Monitoring	Basic monitoring	Advanced monitoring possible
Functionality	Everything included	Everything included