

KDD042 - Mobile Application Framework Selection

Status	Approved
Owner	WENNINGER-ext, Sascha
Stakeholders	PETTIFORD-ext, owen

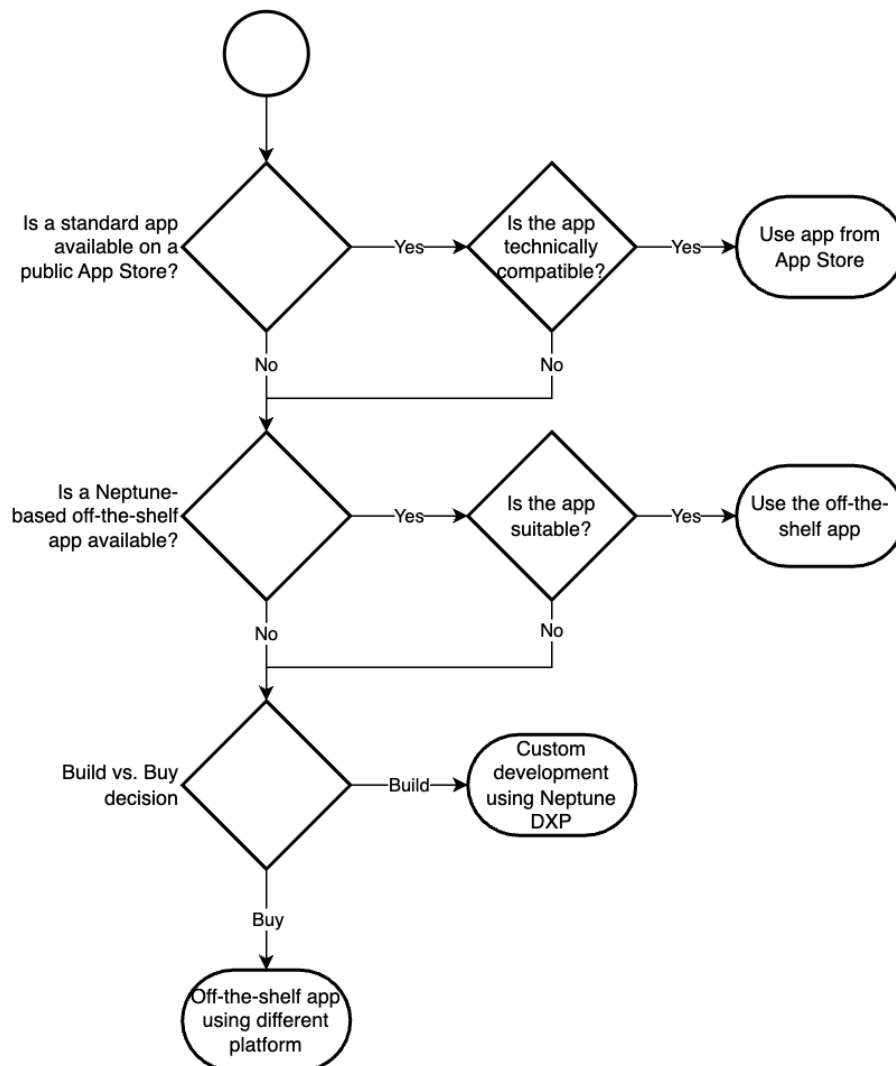
Issue

Several processes at Syensqo benefit from the use of mobile devices either as a core part of a business process, or to provide an alternative interaction mechanism to Desktop PCs for convenience purposes. Typically several different options exist for the implementation of mobile access mechanisms and SAP do not provide a generic/native standard that caters for all scenarios as a default. This document seeks to provide a decision-making framework for the choice of mobile application or mobile application platform to be used.

Recommendation

Where no suitable pre-built application is available on a public App Store, solutions using the Neptune DXP platform are preferred over alternatives. Commercial off-the-shelf applications developed by Neptune partners are preferred over in-house developed applications provided a reasonable functional fit is attainable, and the applications offer extensibility mechanisms compatible with [Clean Core principles](#). Should this not be available, a "build vs. buy" decision should be evaluated using Neptune DXP as the preferred development platform for the custom-development option. This recommendation thus embodies an explicit preference for the use of the Neptune platform due to its proven track record at Solvay/Syensqo, and in recognition of economies of scale in licensing, development, and support. The principle of Simplicity also guides this recommendation by limiting the number of development platforms in concurrent use.

The following decision tree summarises the decision-making process:



Background & Context

Syensqo has experience in the deployment and management of mobile devices and applications in a range of contexts. For example, specialised industrial mobile devices (i.e. ruggedised Zebra devices running Android OS) are used across several GBUs for the Nereid solution for inventory management, while both off-the-shelf and custom-developed mobile apps are available for Syensqo-owned and employee-owned devices through the Ivanti [mobile device management solution](#). Many of the enterprise systems planned to be deployed by the ERP Rebuild program also provide mobile interfaces either via native mobile applications available in public app stores (e.g. for Concur and SuccessFactors), or mobile-capable web UIs.

Assumptions

- It is not possible, nor is it desirable from a usability perspective, to have a single mobile app through which all users access all of the functionality provided by the ERP Rebuild program.
- Although [browser-based UIs are the main method](#) of interacting with systems in scope of ERP Rebuild, the limitations of mobile browsers and web technologies mean browser-based UIs cannot be the sole access mechanism for mobile users. In some situations, e.g. for warehousing applications, native integration with hardware features such as barcode scanners might be required, while some SaaS applications will provide native applications as the sole supported mobile UI.
- Both Android and iOS are considered to be primary mobile device operating systems with which native apps must be compatible.
- For Android, reliance on Google services should be minimised or eliminated. Google services are generally blocked in China. Although devices connected to Syensqo's internal network [can communicate with Google's services](#), run-time reliance on Google services may limit the ability of apps to be used outside of the Syensqo network in China (e.g. while using a mobile internet connection).
- Syensqo's mobile use cases are internal and employee-facing, rather than provided for the general public.
 - At the time of writing, Syensqo does not publish any apps on the Apple iOS App Store or the Google Play Store for the use by customers, partners, suppliers, etc.
 - Use cases do not exhibit the graphical, latency, and performance requirements which are typically met through the use of native development tools (e.g. gaming apps coded in Swift or Java).
 - Apps do not need to pass the quality games imposed by Google and Apple for apps published to their respective public app stores.
- Mobile applications can be pushed to Syensqo-managed devices using [Mobile Device Management](#) regardless of how they were developed.

Constraints

- The decision-making process is primarily concerned with mobile applications interacting with applications from the SAP ecosystem. Neptune DXP does support non-SAP backends as well via the [Open Edition](#) of the product. Being built using Node.js and SQLite or SQLServer, it can be deployed on a wide range of application hosting platforms including on SAP BTP if required.
- The decision tree precludes the use of SAP's own mobile app development platform. At the time of writing, this does not deliver any capabilities which cannot otherwise be obtained via Neptune DXP, nor any commercial/licensing advantages.

Impacts

- Syensqo should continue to invest in the maintenance of technical skills for the Neptune DXP product, including in skilled UI5 developers.
- The application of this decision tree makes it unlikely that native mobile programming skills (e.g. in the Swift language) will be required.
- It may be possible to reuse existing Neptune developments in the future ERP landscape.
- Beyond the ERP Rebuild program, ongoing enforcement of this decision tree via IT Architecture governance is required.

Business Rules

N/A - this document is primarily concerned with technology choices and does not impose business rules.

Options considered

Option A: Use SAP Mobile Services as the primary platform

[SAP Mobile Services](#) is an umbrella platform composed of a number of distinct but related technologies integrated via runtime and management tooling on SAP BTP:

- SAP Mobile Development Kit - a low-code toolkit for development hybrid mobile apps
- Native SDKs for iOS and Android - libraries, extensions, and building blocks for native app development on iOS (using Swift) and Android (using Java), deployed as add-ons into the respective native development tools
- BTP-based backend tools which act as the server-side components for mobile applications, providing OData APIs for apps to interact with data, communicate to back-end systems such as S/4HANA, etc.

In this option, SAP's own mobile development platform is used as the primary platform for mobile apps in the future ERP Rebuild landscape. Many of SAP's own mobile applications such as [SAP Service and Asset Manager](#), are built on or migrated to this platform. Of note is that SAP's strategy in this domain has been quite fluid over the past decade; and that various different products and platforms have at times been designated as SAP's preference, from NetWeaver Mobile, to platforms acquired via Sybase, Syclo, and AppGyver. As of 2024, SAP Mobile Services is also the second iteration of a PaaS option which is hosted by SAP.

Option B: Use Neptune DXP as the primary platform

Neptune is a Norwegian company focused on mobile-enabling SAP systems via a development platform that is tightly integrated into SAP's ABAP stack and aims to minimise the learning curve for SAP developers. The [core product](#) has remained consistent since the company's founding in 2011, and is focused on hybrid mobile app development (rather than fully native apps). Reuse of functionality, code and data models provided by SAP systems is a priority, including use of the same UI5 front-end libraries also used by SAP Fiori. Over time, Neptune's offering was extended to non-SAP backend systems via the [Open Edition](#) that can be deployed outside of an SAP stack and was designed to interact with non-SAP systems, but provide a similar developer experience and tooling. Neptune provides a library of pre-built modules to speed up development of complex but business-agnostic functions, such as offline data sync, authentication, barcode scanning, etc. A rich partner ecosystem provides a significant number of [pre-built applications](#) running on the Neptune platform to address specific business requirements. These applications can also be extended using the Neptune platform.

Option C: Avoid designating a primary platform

In this option, the designation of a primary target platform is avoided, leaving this instead to individual implementation projects to choose based on their individual requirements and available choices at the time. This option prioritises functional fit and implementation convenience over architecture concerns, and acknowledges that a multitude of mobile development platforms will co-exist at Syensqo.

Evaluation

	Option A SAP Mobile Services	Option B Neptune DXP	Option C Best-of-breed
Strategic	● Strategic choice of SAP. Future development effort by SAP into its own mobile offerings will be focused on SAP Mobile Services. ➖ SAP's strategic choice has changed significantly at least 4 times in the preceding decade as successive acquisitions redirected its focus for both front-end and back-end development approaches and tooling. This has frequently caused SAP's own first-party offerings to be sunset and replaced with different re-implementations of the same features, but usually with licensing impacts.	➕ Consistent product strategy and direction for over 10 years which has allowed its partner ecosystem to develop significant IP and evolve their products over years to be extremely competitive against SAP's first-party offerings. ● Neptune is backed by private equity. Should the company be sold to a larger software company, its focus on the SAP and partner ecosystem and its commercial models may be impacted.	➖ Deploying a variety of different mobile platforms will prevent the creation of significant economies of scale in the extension and operations at Syensqo.
Functional fit	➖ Limited to existing solutions by SAP, or custom development. An excellent off-the-shelf mobile app using a different framework would not be deployed.	➖ Limited to existing solutions by Neptune partners or custom development. An excellent off-the-shelf mobile app using a different framework would not be deployed.	➕ Potential to achieve best-possible outcomes for end users by not constraining the available applications to only those supported on a particular platform.
Simplicity	➖ Mandatory use of BTP components for development and runtime components means a comparatively higher degree of complexity at runtime. ➖ Availability of both native and hybrid development paradigms means different code bases are likely needed for different operating systems.	➕ Low overall architectural complexity as the SAP Edition of Neptune DXP runs inside the S/4HANA system. ➕ Singular development paradigm focused on hybrid mobile apps means that iOS and Android can be supported by the same code base.	➖ Results in the concurrent deployment of multiple different mobile application frameworks for development and runtime ➖ Greater operational complexity as multiple different components have to be monitored and maintained.

Implementation cost	➕ The use of BTP as a runtime for the mobile back-end avoids the need to install additional systems, making this a serverless architecture. ➖ Prior experience has shown that SAP's first-party offerings are generally more expensive than solutions developed by Neptune partners. This has also been confirmed via budgetary pricing analysis for EAM use cases, where SAP's offering was almost 3 times as expensive.	➕ With the SAP Edition of Neptune DXP running inside the S/4HANA system as an add-on, implementation cost is reduced by not requiring additional system components. ➕ Both the Neptune platform and partner-developed applications running on it appear to be significantly cheaper than SAP's first-party offerings based on initial analysis (e.g. for EAM). ➕ Good availability of skilled Neptune developers (the Neptune developer community has close to 5000 members), and relatively easy upskilling for SAP developers with Fiori experience.	➖ Implementing a range of different platforms will make it more difficult to build economies of scale in platform and user licensing. ➖ A variety of skills needed to extend and customise a range of different applications, or several different vendors would need to be engaged for the work.
Agility	➕ Reusing an existing platform means greater agility through reuse of vendor due diligence checks and platform functions such as authentication, patching, etc.	➕ Reusing an existing platform means greater agility through reuse of vendor due diligence checks and platform functions such as authentication, patching, etc.	➕ Ability to choose best platform for any given use case. ➖ Requirement to execute commercial onboarding, cybersecurity validations, and other due diligence checks for each platform decreases agility.
Maintenance and Administration	➕ SAP Mobile Services / BTP standard should align neatly with overall SAP system maintenance and administration approaches ➕ A broad depth of skilled talent available to support avoiding the need for highly specialised recruitment to support and maintain BTP deployed solutions.	➕ While not as centrally adopted as SAP native, Neptune product is built to align with SAP development standards ➕ Given alignment to SAP standards, suitably skilled resources to support are available with minimal product training required.	➖ Implementing a range of different platforms means multiple skills / vendors required to support bespoke solutions ➖ Increased risk of creation of technical debt with solutions deployed not aligning to any standard.

See also

[Mobile Device Management](#)

[Network architecture for China](#)

[KDD036 - User Access to Enterprise Systems](#)

[KDD025 - Mobility Solutions in Logistics and Warehouse Management Processes](#)

[KDD016 - Mobility Solution Direction for Enterprise Asset Management](#)

[Explanation of Clean Core](#), by SAP Learning

File	Modified
PDF File Workspace Mail - For Approval_ KDD042 - Mobile Application Framework Selection.pdf	Oct 02, 2024 by FALL-ext, Cheikh
File Mobile App Decision-making Process.drawio	Aug 23, 2024 by WENNINGER-ext, Sascha

[Download All](#)

Change log

Version	Published	Changed By	Comment
CURRENT (v. 12)	Oct 08, 2024 21:09	WENNINGER-ext, Sascha	
v. 11	Aug 23, 2024 12:02	WENNINGER-ext, Sascha	
v. 10	Aug 23, 2024 11:48	WENNINGER-ext, Sascha	

v. 9	Aug 23, 2024 11:32	NARAHARI-ext, Bhargavi
v. 8	Aug 23, 2024 09:43	WENNINGER-ext, Sascha
v. 7	Aug 22, 2024 10:39	DA VINCI-ext, Daniel
v. 6	Aug 22, 2024 10:35	DA VINCI-ext, Daniel
v. 5	Aug 22, 2024 10:14	DA VINCI-ext, Daniel
v. 4	Aug 22, 2024 06:27	WENNINGER-ext, Sascha
v. 3	Aug 22, 2024 06:20	WENNINGER-ext, Sascha

[Go to Page History](#)

Workflow history

Title	Last Updated By	Updated	Status
There are no pages at the moment.			

Workflow history

This view shows the 5 most recent entries. The complete workflow log is available from the 'Document Activity' menu item.

Oct 08, 2024	Actor	Type	Activity	Version
Approved	 WENNINGER-ext, Sascha	Edit	updated the page at 9:09 pm	
		State	changed state to Approved at 7:09 pm	v12
Edited following Approval	 WENNINGER-ext, Sascha	State	gave <i>Minor update</i> approval at 7:09 pm	
		State	changed state to Edited following Approval at 7:09 pm	v12
Oct 01, 2024				
Approved	 PETTIFORD-ext, owen	State	changed state to Approved at 7:51 pm	v11
Pending SteerCo Review	 PETTIFORD-ext, owen	State	gave <i>Final Approval</i> approval at 7:51 pm	
		State	changed expiry date to '15 Oct, 2024 07:50 pm' at 7:50 pm	
		State	changed state to Pending SteerCo Review at 7:50 pm	v11
Pending Stakeholder Review	 PETTIFORD-ext, owen	State	gave <i>Stakeholder Review</i> approval at 7:50 pm	
			Correct SQLite to MS SQL Server - otherwise OK	