

Projektauftrag

SABIS-API & SABIS-App

Implementierung einer Schnittstelle (API) für die Anbindung von Drittapplikationen an SABIS sowie Implementierung einer App für SABIS.

PORSCHE
INFORMATIK GMBH

Louise-Piëch-Straße 9
A – 5020 Salzburg

Stand: 08.04.2020
Version: 02:36 Uhr

Internal

Inhalt

1	Kurzbeschreibung	3
2	Organisation	3
3	Berichtswesen an Auftraggeber	3
4	Umfang	3
5	Risikobetrachtung	3
6	Terminplan	3
7	Budget	4
8	Verfügbarkeit und Betrieb	5

1 Kurzbeschreibung

Implementierung einer Schnittstelle (API) zur Anbindung von Drittapplikationen an SABIS. Des Weiteren wird eine SABIS-App umgesetzt, um einen zusätzlichen Kommunikationskanal für Teilnehmer und Trainer zu schaffen. Beide Anforderungen bilden zugleich eine Vorstufe für einen Technologiewechsel in der SABIS Infrastruktur.

2 Organisation

Rolle	Auftraggeber	Auftragnehmer
	Porsche Austria GmbH, Porsche Bank GmbH	Porsche Informatik GmbH Louise-Piëch-Straße 9 5020 Salzburg
Projektleiter	Alfred Kühnel	Robert Behmüller
Projektmitarbeiter	Barbara Rückl, Dominik Prock, Georg Struber, Hannes Guttmann, Julia Speil, Martina Jäger, Neubauer Tanja	Ganna Shulika Gustav Hlavac Ingrid Oberbuchlerova
Entscheidungsinstanz	Peter Oberweger Leiter Adrian	Gebhard Dunst

3 Berichtswesen an Auftraggeber

Monatliche Meetings der Projektleitungen und des Entwicklungsteams. Zweimonatliche Meilensteinmeetings des gesamten Projektteams (siehe Projektmitarbeiter) jeweils für die SABIS-API sowie für die SABIS-App.

4 Umfang

4.1 Zielsetzung

- 1.) Umsetzung einer Anbindungsmöglichkeit an SABIS mittels einer Schnittstelle (API). Im Zuge des Projektes wird eine Auslesemöglichkeit des SABIS Kernsystems umgesetzt.
- 2.) Schaffung eines zusätzlichen Kommunikationskanals für Kursteilnehmer, Schulungsverantwortliche und Kurstrainer via einer App für SABIS. Zur Umsetzung der „Teilnehmerliste“ wird die generelle Auslesemöglichkeit um eine Schreibfunktionalität eines neuen Status erweitert werden, dies ist der Status „abwesend“ für Teilnehmer.

Der Funktionsumfang für die Ansicht Schulungsverantwortlicher in der App ist noch zu definieren.

4.2 Durchgeführte Vorarbeiten

Es wurde ein Konzept für die technische Architektur und Umsetzung der SABIS-API & SABIS-App ausgearbeitet. Weiters wurden im Zuge einer Bachelor Arbeit in Kollaboration mit der Universität Salzburg Screen Mockups erstellt und anhand dieser Mockups die Anforderungen für die SABIS-App spezifiziert.

4.3 Im Projekt

4.3.1 SABIS-API

Im Zuge des Projektes wird eine Schnittstelle für die Anbindung von Drittapplikationen umgesetzt. Diese Schnittstelle teilt sich wesentliche Infrastrukturkomponenten mit der SABIS-App (siehe Abbildung 1). Dieser Synergieeffekt hilft Doppelentwicklungen zu vermeiden und die SABIS-API als auch die SABIS-App profitieren gemeinsam von zukünftigen Funktionalitäten.

Im Zuge des Projektes wird eine Auslesemöglichkeit des Kernsystems SABIS umgesetzt werden. Zusätzlich zu dieser Auslesemöglichkeit wird das Rückschreiben eines neuen Status „abwesend“ umgesetzt, siehe Requirement [SA3REQ-203](#).

Die Umsetzung der SABIS-API soll eine möglichst generalistische Schnittstelle bieten, die eine Anbindung für ein breites Spektrum an Drittapplikationen möglich macht. Die Schnittstelle wird im Requirement [SA3REQ-226](#) genau spezifiziert und bestätigt.

4.3.2 SABIS-App

Grund-App in Form einer Progressive Web App (PWA), es handelt sich dabei um eine HTML5 Browser Anwendung, die sich für den Benutzer wie eine klassische App aus dem Appstore verhält, aber plattformunabhängig entwickelt ist und unabhängig von den Appstores (Google Play / Apple Store) zu verwenden ist.

4.3.2.1 Authentifizierung via PartnerNet ([SA3REQ-202](#))

Die Authentifizierung der Benutzer wird über Partner.net erfolgen mittels OAuth2.

Benötigte Informationen über Kursteilnehmer sowie Firmeninformationen werden über den von Partner.net bereitgestellten Service „Data-API“ bezogen werden.

4.3.2.2 Datenanbindung an SABIS Kernsystem ([SA3REQ-203](#))

Auslesemöglichkeit des Kernsystems mit Rückschreibemöglichkeit eines neuen Status „abwesend“ für Teilnehmer.

4.3.2.3 Kursbewertung/Kursfeedback ([SA3REQ-204](#))

Funktionalität zum Bewerten von Kursen an denen man teilgenommen hat.

4.3.2.4 Push-Benachrichtigungen für Teilnehmer (vorläufig nur Android) ([SA3REQ-205](#))

Funktionalität für Push Benachrichtigungen um Kursteilnehmer über bevorstehende Termine sowie Microlearnings zu informieren. Aus technischen Gründen ist derzeit nur eine Umsetzung auf Android vorgesehen.

4.3.2.5 Kursteilnehmerübersicht ([SA3REQ-207](#))

Die Teilnehmerliste soll die Möglichkeit für Trainer beinhalten abwesende Kursteilnehmer zu markieren. Dieser Status wird in das SABIS Kernsystem zurückgeschrieben. Für Kursteilnehmer soll ebenfalls eine abgeänderte Form der Kursteilnehmerübersicht implementiert werden, von dieser Übersicht soll der „Kurs-Chat“ schnell erreichbar sein um z.B. Fahrgemeinschaften organisieren zu können.

4.3.2.6 Kursterminübersicht ([SA3REQ-208](#))

In der Kursterminübersicht sollen alle bevorstehenden oder laufenden Kurstermine für den Benutzer angezeigt werden. Detailinformationen über den Kurstermin können ebenfalls abgerufen werden, unter anderem ein Google-Maps Link der Auskunft über den Kursort liefert. Auch für Schulungsverantwortliche relevant, sprich eine Ansicht aller Personen zu einer BNR, die auf Schulungen angemeldet sind bzw. auf einer Warteliste gebucht sind.

4.3.2.7 Kursübersicht ([SA3REQ-206](#))

Die Kursübersicht soll Kursteilnehmern eine Plattform bieten um Kursinformationen zu erhalten. Es sollen die bevorstehenden Kurse, laufenden Kurse, Wartelisten sowie bereits abgeschlossenen Kurse angezeigt werden.

4.3.2.8 Auflistung der Benutzer Aktivität ([SA3REQ-224](#))

Im „User Log“ sollen Benutzeraktivitäten die in der SABIS-App durchgeführt wurden angezeigt werden, wie z.b.: eine Namensänderung oder Änderungen im Profil.

4.3.2.9 Chat ([SA3REQ-210](#))

Die Chatfunktion soll einerseits Chaträume für bevorstehende sowie laufende Kurse bieten. Ein abgeschlossener Kurs soll nach 3 Monaten, genauer Zeitraum ist noch durch den Auftraggeber zu definieren, archiviert werden und nur noch „read-only“ den Teilnehmern zur Verfügung stehen.

Des Weiteren soll die Kommunikation zwischen Benutzern über den Chat möglich sein, Stichwort „direct Messaging“. Das Einfügen von Dokumenten, Fotos & Videos in den Chat soll möglich sein. Vom User gelesene Nachrichten sollen als „gelesen“ markiert werden.

Die Umsetzung der Chat Funktionalität wird zusammen mit den Chief Technical Architects (CTAs), Demand Management sowie dem carLOG Team dahingehend analysiert ob eine Zukaufslösung bzw. eine gemeinsame Implementation mit carLOG („Chat Modul“) zum Tragen kommen soll.

4.3.2.10 Microlearnings mit Gamification ([SA3REQ-209](#))

Über die Microlearning Plattform sollen den Kursteilnehmern Lerninhalte vorab zur Verfügung gestellt werden (mit Videos). Diese Lerninhalte können ebenfalls mit Gamification ausgestattet werden, z.b.: ein Quiz das mit zeitlicher Beschränkung vom Kursteilnehmer durchgeführt werden kann. Die maximal möglichen Versuche um eine Gamification durchführen zu können soll ebenfalls einstellbar sein (z.b.: 3 Versuche pro Quiz). Der Abschluss einer Gamification vom Kursteilnehmer soll einen „Reward“ in Form von Punkten (Scoring) auslösen.

Der Kursteilnehmer soll auf anstehende Microlearnings sowie deren Gamification mittels Push Benachrichtigung aufmerksam gemacht werden.

Für den Kursleiter (Trainer) und für Schulungsverantwortliche soll eine Micro-Learning Übersicht mit den Gamification Scores der Kursteilnehmer bei bevorstehenden sowie laufenden Kursen zur Verfügung stehen.

Die Wartung der Microlearning & Gamification Inhalte wird über eine neue Wartungsoberfläche umgesetzt werden. Diese Wartungsoberfläche wird über einen Menüpunkt im Partner.net zugänglich sein.

4.3.2.11 Score Übersicht (Gamification Aspekt bei Microlearnings) ([SA3REQ-225](#))

Die Score Übersicht soll den Benutzern und dem Schulungsverantwortlichen im Betrieb Auskunft über ihre erworbenen Punkte bei der Gamification der Microlearnings geben. Angezeigt wird der gesamte Punktestand sowie der „High-Score“ der letzten Gamifications.

Die genaue Spezifikation der Funktionalitäten wird anhand von Mockups durchgeführt und in den Requirements, siehe Aufzählung oben, dokumentiert und bestätigt.

SABIS-API & SABIS-App

4.3.3 Technische Architektur SABIS-App & SABIS-API

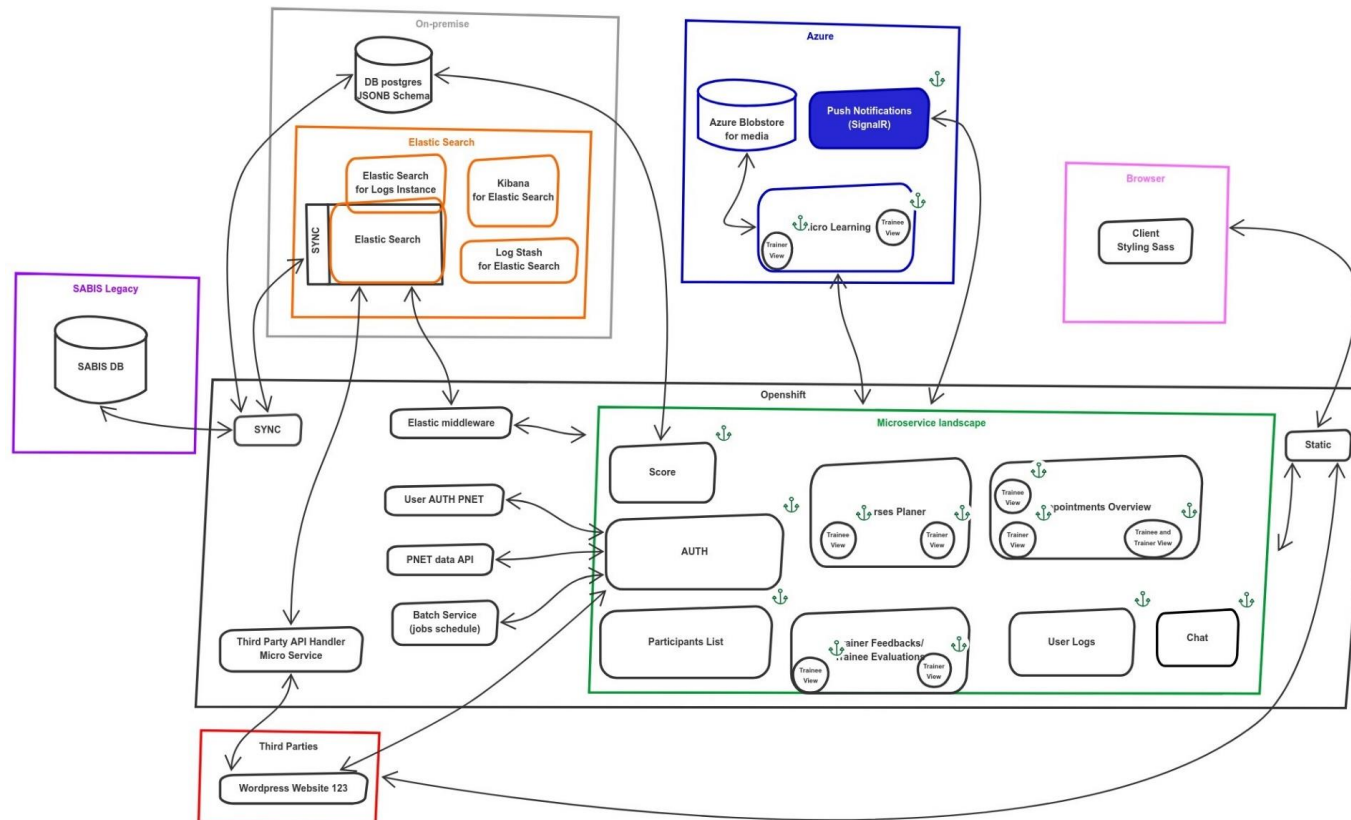


Abbildung 1: Konzept der techn. Architektur für SABIS-App sowie SABIS-API.

Implementierung einer App für SABIS

4.4 Nicht im Projekt

4.4.1 SABIS-App

Push-Benachrichtigungen für IOS (derzeit fehlt die Unterstützung seitens Apple, ist aber für kommende IOS Updates angekündigt und dann mit wenig Aufwand integrierbar.)

- Microsoft Teams-Integrationen
 - Auf eine MS Teams Integration wird im Projekt aufgrund technischer Rahmenbedingungen (Installation von Teams auf dem Endgerät notwendig) abgesehen

Die Umsetzung einer „native-App“ ist im Projekt nicht enthalten.

Der Source-Code wird nicht an den Auftraggeber ausgeliefert. Der Source-Code wird bei der Porsche Informatik mit Hilfe einer Versionsmanagement-Umgebung verwaltet.

4.4.2 SABIS-API

Die in diesem Projekt geplante Umsetzung der SABIS-API wird hauptsächlich eine Auslesemöglichkeit des SABIS Kernsystems ermöglichen. Funktionalitäten, die eine Schreibmöglichkeit in das Kernsystem voraussetzen, wie z.B. das Buchen eines Kurses, sind im Projekt nicht vorgesehen. Das Projekt legt allerdings das Fundament um zukünftige Erweiterungen der Schnittstelle, auch in Hinblick einer Schreibmöglichkeit, zu erlauben.

Der Source-Code wird nicht an den Auftraggeber ausgeliefert. Der Source-Code wird bei der Porsche Informatik mit Hilfe einer Versionsmanagement-Umgebung verwaltet.

4.5 Leistungen des Auftraggebers

- Unterstützung bei der Abstimmung und Definition von fachlichen Themen
- Berücksichtigung der im Projekt budgetierten Aufwände bei der Detaildefinition von Features
- Unterstützung bei System- und Abnahmetests
- Abnahme von einzelnen Entwicklungspaketen
- Einschulung bzw. „Vermarktung“ an die Anwender
- Verantwortung hinsichtlich der Einhaltung und Umsetzung von Datenschutz-Themen

4.6 Leistungen von Dritten

Ggfs. Unterstützung bei Webdesigns (im Rahmen des vorhandenen Budgets).

5 Risikobetrachtung

Siehe Risikomonitor für Initiales Risikomanagement und laufendes Risikomanagement.

6 Terminplan

Arbeitsschritt	Beschreibung	Verantwortlich	Termin
Projektstart			01.08.2019
Kickoff			August 2019
Meilenstein 1	Etablierung der Entwicklungsumgebungen	SHUG, HLAV, OBIN	01.10.2019
Meilenstein 2	Erste testbare Version, Fertigstellung erster Microservice	SHUG, HLAV, OBIN	30.11.2019
Meilenstein 3	Infrastruktur und Umgebungen etabliert (DEV – QA – PROD)	SHUG, HLAV, OBIN	28.02.2020
Meilenstein 4	Feature Complete	SHUG, HLAV, OBIN, BEH, LAMI	01.05.2020
Meilenstein 5	Pilotierung	SHUG, HLAV, OBIN, BEH	01.06.2020
Meilenstein 6	Go-Live	SHUG, HLAV, OBIN, BEH	01.08.2020
Projektende		SHUG, HLAV, OBIN, BEH	31.10.2020

Добавлено примечание ([ВЕН1]): Michal Lange mitverantwortlich für die Umsetzung des Datenabgleich

6.1 Änderungen während des Projektes

Sollten sich der Funktionsumfang bzw. die Anforderungen gegenüber dem definierten Projektumfang verändern, ist zu prüfen, welchen Einfluss diese Veränderungen auf den Projektverlauf haben.

Erweiterungen, die eine Termin- oder Kostenänderung bewirken, sind vom Auftragnehmer aufzuzeigen und von der Entscheidungsinstanz beim Auftraggeber zu genehmigen.

6.2 Abnahme

Die Abnahme des Projektes erfolgt vom Projektbeauftragten des Auftraggebers bzw. vom Projektauftraggeber über die ProjektverwaltungsDB in Notes.

Weiterentwicklungen müssen anhand eigens definierter Projekte oder Kleinaufträge über die Produkt-Anforderungsdatenbank angefordert werden.

7 Budget

7.1 Einmalige Kosten

Der derzeit gültige Stundensatz der Porsche Informatik beträgt € 85,--.

7.1.1 Projektkosten

Der Projektaufwand beträgt **4362,75** Stunden. Das ergibt eine Auftragssumme von € 370.833,75. Die Verrechnung erfolgt laufend monatlich. Zirka **1160** Stunden (€ 98.900,-) können davon im Wirtschaftsjahr 2019 geleistet werden und sind auch budgetiert. Dies würde einen Restaufwand von **3202,75** Stunden (€ 272.233,75) für das Wirtschaftsjahr 2020 ergeben. Für das Wirtschaftsjahr 2020 wird eine neue Kostenverteilung von der Group-IT zusammen mit den Fachbereichen definiert werden.

Der Projektaufwand gliedert sich in folgende Positionen:

Position	Stunden
SABIS-API	2000,78
SABIS-App	2361,98

Die Verrechnung erfolgt nach folgendem Verteilungsschlüssel für das Wirtschaftsjahr 2019:

Firma	Kostenstelle	Anteil in %	Betrag in €
PBV	20800	20	74166,75
POA STT	46800	40	148333,5
POA STO	47020	40	148333,5

Добавлено примечание ([ВЕН2]): Hier habe ich einiges geändert

7.2 Laufende Kosten

Die Porsche Informatik übernimmt nach Projektende die Wartung des Produktes im Rahmen des vereinbarten Wartungsbudgets.

Leistung	Kosten
Laufende Betriebskosten	5000.- p.a.
Software-Wartung pro Jahr (Pauschale)	100h p.a. (ab 2020)
POI Service Desk	Verrechnung laut Ist-Aufwand
POI Support durch Serviceteams	Verrechnung laut Ist-Aufwand

Добавлено примечание ([ВЕН3]): Dummy Wert, finale Antwort von Manfred noch ausständig. Ich frage regelmäßig bei ihm nach.

Добавлено примечание ([ОВЕР4]): Muss sich ähnlich der heutigen Kosten bewegen

8 Verfügbarkeit und Betrieb

Es gilt die Verfügbarkeit und das Service Level Agreement von Sabis (SLA Klasse „C“).

Der Support nach Projektende erfolgt nach folgender Tabellenstruktur:

Support Level	Verantwortlich	Beschreibung
1	Porsche Informatik	Service Desk: Eskaliert an Group-IT (Alfred Kühnel, Elisabeth Pesendorfer, Peter Oberweger)
2	Group-IT (eskaliert an Service Teams oder Team ASMS Entwicklung)	Fachliche Fragen: Alfred Kühnel, Elisabeth Pesendorfer, Peter Oberweger
3	Porsche Informatik	Team ASMS SABIS-App Entwicklungsteam