Test specification E2E OAT

TC-ID	Testcase	Description	Manual test steps			
		Open the WebApp in Browser		Schritt	Eingabe/Daten	Erwartete Ergebnisse
TXR-2013	INT_IssApp_Start_WebApp		1	Open Browser	https://dgca-issuance-web.cfapps.eu10.hana.ondemand.com/index.html/	WebApp is starting
		Insert relevant Data in Issuer App. Send inserted Data to national backed.		Schritt	Eingabe/Daten	Erwartete Ergebnisse
			1	open the data entry mask		Data Entry Mask is shown
			2	insert Family name in textfield "Family name"		"Family name" is shown in textfield
			3	insert given name in textfield "Given name"		"Given name" is shown in textfield
			4	insert date of birth in textfield with picker 'Date of Birth'		date is shown in textfield
			5	insert "Disease/Agent*" in textfield Disease/Agent*		"Disease/Agent*" is shown in textfield
			6	choose vaccination type in combo box 'Vaccine/Prophylaxis*'		vaccination type is shown textfield
			7	choose medical product in combo box 'Medicinal Product*'		medical product is shown in textfield
	INT_IssApp_Create_QR-Code		8	choose Organisations Management System* in combo box 'Organisations Management System*'		Organisations Management System is shown in textfield
			9	insert dose number in Textfield "Dose Number*"		dose number is shown in textfield
			10	insert total series of doses in Textfield "Total Series of Doses*"		total series of doses is shown in textfield
			11	insert vaccination date in textfield with picker'vaccination date'		vaccination date is schown textfield
			12	choose Issuer country in combo box 'Issuer Country*'		issuer country is shown in textfield
			13	insert certificate issuer in textfield "Certificate Issuer*"		Certificate Issuer is shown in textfield
TXR-2017			14	push "next" button		QR-code will be generated with inserted data
		send unsigned QR-code to national Backend, which signs it and send it back to the Issuer App. The signed QR-code will be displayed on screen		Schritt	Eingabe/Daten	Erwartete Ergebnisse
TXR-2019	INT_IssApp_Request_signed_QR-Code		1	Send created QR-Code to national backend via "finish process" button		QR-Code will be send - national backend returns signed QR-Code
		print the QR-code/vaccination certificate with included print service(?)		Schritt	Eingabe/Daten	Erwartete Ergebnisse
	INT_IssApp_Print_signed_QR-Code		1	open signed QR-code		signed QR-code will open
TXR-2020			2	push the print-button		signed QR-code will be printed
		scann the QR-code with the wallet app		Schritt	Eingabe/Daten	Erwartete Ergebnisse
	INT_WalletApp_Citizen_scans_QR-Code		1	open the (internal) QR-codescanner		QR-codescanner starts
TXR-2028			2	position the QR-Code under the camera		QR-code is displayed sharply
		show the saved certificate on mobile device within the details of the data		Schritt	Eingabe/Daten	Erwartete Ergebnisse
	INT_WalletApp_shows_the_certificate_on_mobile_device		1	open the internal storage		all scanned QR-codes will be listed
TXR-2029			2	Choose one QR-code		QR-code will be displayed on screen
		on Start the WalletApp a biometric request has to start. To be sure that a verified person get access to WalletApp Data		Schritt	Eingabe/Daten	Erwartete Ergebnisse
	INT_WalletApp_biometric_security		1	open WalletApp on Mobile Device		biometric data are requested
TXR-2032			2	scan your biometric data		WalletApp starts
	INT_WalletApp_negative_biometric_securit	on Start the WalletApp a biometric request has to start. To be		Schritt	Eingabe/Daten	Erwartete Ergebnisse

	INT_WalletApp_negative_biometric_security	sure that a verified person get access to WalletApp Data	1	open WalletApp on Mobile Device		biometric data are requested
TXR-2033			2	scan wrong biometric data		Error: Access denied
	INT_VERIAPP_verify_qr_code_for_a_valid_dgc	A Digital Green Certificate with: 1) a valid QR Code; 2) valid Payload; 3) valid Attributes. is presented for offline verification. The Verifier App confirms that the DGC is valid. It also tests that the same DGC can be verified twice by the same VeriApp instance.		Schritt	Eingabe/Daten	Erwartete Ergebnisse
			1	VeriApp scans QR-Code.		QR-Code is approved as verified.
			2	VeriApp scans the same QR-Code for a second time.		QR-Code is again approved as verified.
TXR-2075						
	INT_VERIAPP_neg_verify_qr_code_with_invalid_signature	A Digital Green Certificate (DGC) with invalidly signed QR-Code is presented for offline verification. The Verifier App evaluates the DGC as invalid. An Error Code for "Invalid Signature" is shown.		Schritt	Eingabe/Daten	Erwartete Ergebnisse
			1	VeriApp scans QR-Code.		The VeriApp evaluates the DGC as invalid. An Error Code for "Invalid Signature" is shown.
TXR-2077						
	INT_VERIAPP_neg_verify_qr_code_with_invalid_payload_syntax	A Digital Green Certificate with correct signature but syntactically invalid payload (e.g. missing name etc.) is presented for verification. The signature is validated but the DGC is evaluated as invalid due to invalid Payload. An Error Code "Invalid Payload" is returned.		Schritt	Eingabe/Daten	Erwartete Ergebnisse
			1	VeriApp scans QR-Code.		QR-Code signature is approved as valid.
TXR-2079			2	VeriApp reads payload.		The DGC is evaluated as invalid. An Error Code "Invalid Payload" is shown.
	INT_VERIAPP_render_dgc_for_type_PCRtest	A validly signed Digital Green Certificate of type (PCR) TEST is presented for verification. The testcase tests presentation of the DGC Data for the DGC of type test, independently of test result (postive or negative).		Schritt	Eingabe/Daten	Erwartete Ergebnisse
			1	VeriApp scans QR-Code.		The DGC is approved as valid and the Contents Data is presented for type TEST. The content is preseneted as a positive or negative quick test.
TXR-2084						
	INT_VERIAPP_render_dgc_for_type_vac	A valid Digital Green Certificate of type VAC (owner has been vaccinated) is presented for verification. The testcase tests presentation of the DGC Data.		Schritt	Eingabe/Daten	Erwartete Ergebnisse
TXR-2085			1	VeriApp scans QR-Code.		The DGC is approved as valid and the Contents Data is presented for type VAC.
	INT_VERIAPP_render_dgc_for_type_rec	A valid Digital Green Certificate of type REC (owner has recovered) is presented for verification. The testcase tests presentation of the DGC Data.		Schritt	Eingabe/Daten	Erwartete Ergebnisse
TXR-2086			1	VeriApp scans QR-Code.		The DGC is approved as valid and the Contents Data is presented for type REC.
	INT_VERIAPP_fetch_an_use_manually_triggered	The Verifier App has to support the manual triggering of the synchronisation process.		Schritt	Eingabe/Daten	Erwartete Ergebnisse
			1	The VerifierApp has been installed. Internet connection is available. It has been less than 24 hours since the last synchronisation. The user triggers the synchronisation manually.		A Synchronisation process has been triggered and the keys have been updated.
TXR-2087						
	INT_VERIAPP_fetch_and_use_resynchronise_after_offline_state	This testcase examines the case where no synchronisation has taken place in the last 24 hours due to missing internet connection. As soon as the internet connection is available again, the verifier app should initiate synchronisation.		Schritt	Eingabe/Daten	Erwartete Ergebnisse
			1	The VerifierApp has been installed and at it is has been 24 hours since the installation.		A Synchronisation process has been triggered and the keys have been updated within the last 24 hours.
			2	After the synchronisation has been done, the internet is switched off for at least 24 hours.		No synchronisation of the keys database could take place.
TXR-2088			3	The internet connection is available again.		The verifier app initiates synchronisation (fetch and use) within the next 24 hours.
		The Verifier App has to synchronise its public key database daily		Schritt	Eingabe/Daten	Erwartete Ergebnisse

TXR-2089	INT_VERIAPP_fetch_an_use_daily_synchronisation	with the backend. Internet Connection is available.	1	The VerifierApp has been installed and at it is has been 24 hours since the installation.		A Synchronisation process has been triggered and the keys have been updated within the last 24 hours.
TXR-2094	INT_VERIAPP_render_dgc_for_test_result_positive	A validly signed Digital Green Certificate of type POSITIVE TEST (owner has tested positive) is presented for verification. The testcase tests presentation of the DGC Data.		Schritt	Eingabe/Daten	Erwartete Ergebnisse
			1	VeriApp scans QR-Code.		The DGC is read and a positive test result is displayed.
TXR-2103	INT_WalletApp_register_QR-Code_with_TAN	The QR-code is only allowed to save on one device. Therefor the citizen gets a TAN wich can be used only one time. After the registration, the TAN can't be used twice.		Schritt	Eingabe/Daten	Erwartete Ergebnisse
			1	scann QR-code with integrated barcode-scanner		Barcode will be shown on screen
			2	push save button		TAN will be requested
			3	insert valid TAN		scanned QR-code will be saved
TXR-2105	INT_WalletApp_start_WalletApp_with_PIN	If the citizen has no biometric data on his mobile device it shoul be possible to start the device by PIN		Schritt	Eingabe/Daten	Erwartete Ergebnisse
			1	start the WalletApp on mobile device		biometric data are requested
			2	user push cancel		a user PIN is requested
			3	insert the correct PIN		WalletApp starts
TXR-2106	INT_WalletApp_negative_register_QR-Code_with_TAN_-_TAN_expired	The QR-code is only allowed to save on one device. Therefor the citizen gets a TAN wich can be used only once for a defined time after creation. (Expirationtime has to be defined) After this time, the TAN can't be used anymore.		Schritt	Eingabe/Daten	Erwartete Ergebnisse
			1	scann QR-code with integrated barcode-scanner		Barcode will be shown on screen
			2	push save button		TAN will be requested
			3	insert expired TAN		An error occurred: TAN expired QR-code will not be saved
TXR-2107	INT_WalletApp_negative_register_QR-Code_with_TAN _wice	The QR-code is only allowed to save on one device. Therefor the citizen gets a TAN wich can be used only one time. After the registration, the TAN can't be used twice.		Schritt	Eingabe/Daten	Erwartete Ergebnisse
			1	scann QR-code with integrated barcode-scanner		Barcode will be shown on screen
			2	push save button		TAN will be requested
			3	insert valid TAN a seconed time		an error occurred: TAN can't be used twice
	INT_IssApp_Create_corrected_QR-Code	Insert relevant Data in Issuer App with wrong birthdate. Start creation of QR-code. Get QR-code with wrong birthday. proof data in QR-code and find the misstake. correct birthday in Issuer App and create new QR-code.		Schritt	Eingabe/Daten	Erwartete Ergebnisse
			1	open the data entry mask		Data Entry Mask is shown
			2	insert Family name in textfield "Family name"		"Family name" is shown in textfield
			3	insert given name in textfield "Given name"		"Given name" is shown in textfield
			4	insert wrong date of birth in textfield with picker 'Date of Birth'		date is shown in textfield
			5	insert "Disease/Agent*" in textfield Disease/Agent*		"Disease/Agent*" is shown in textfield
			6	choose vaccination type in combo box 'Vaccine/Prophylaxis*'		vaccination type is shown textfield
			7	choose medical product in combo box 'Medicinal Product*'		medical product is shown in textfield
			8	choose Organisations Management System* in combo box 'Organisations Management System*'		Organisations Management System is shown in textfield
			9	insert dose number in Textfield "Dose Number*"		dose number is shown in textfield
			10	insert total series of doses in Textfield "Total Series of Doses*"		total series of doses is shown in textfield
			11	insert vaccination date in textfield with picker'vaccination date'		vaccination date is schown textfield

			12	choose Issuer country in combo box 'Issuer Country*'		issuer country is shown in textfield
			13	insert certificate issuer in textfield "Certificate Issuer*"		Certificate Issuer is shown in textfield
			14	push "next" button		QR-code will be generated with inserted data
			15	push "next" button		QR-code will be generated with inserted data
			16	check the inserted data		wrong birthday is shown
			17	push "correct patient data" button		inserted data will be shown in data entry mask
			18	edit birthday field and insert correct birtday		corrected birtday is shown
TXR-2113			19	push "next" button		QR-code will be generated with corrected data
	INT_VERIAPP_render_dgc_for_test_result_negative	A validly signed Digital Green Certificate of type Negative TEST (owner has tested negative) is presented for verification. The testcase tests presentation of the DGC Data.		Schritt	Eingabe/Daten	Erwartete Ergebnisse
TXR-2182			1	VeriApp scans QR-Code.		The DGC is read and a negative test result is displayed.
	INT_WalletApp_valid_TAN_which_does_not_belong_to_this_qr-code	Issuer has created two different QR-codes. Each with valid TAN. He gave citizen A qr-code A with with valid TAN to qr-code B. He gave citizen B qr-code B with with valid TAN to qr-code A. So, we have A valid TAN which belongs to an other valid QR-code.		Schritt	Eingabe/Daten	Erwartete Ergebnisse
			1	scann QR-code with integrated barcode-scanner		Barcode A will be shown on screen
			2	push save button		TAN will be requested
			3	insert valid TAN B which does not belong to this qr-code (dgci)		TAN B will be accepted by wallet app
			4	send data to national backend		national backend will proof the data and returns an error to wallet app
TXR-2187			5	get error code from national backend		qr-code will not be saved
	INT_WalletApp_use_deep_link_by_SMS_to_import_certificate	Optionally you can use a deep link instead of a 2D Code to initiate the certificate import in the wallet app. The deep link can look like: dgc://example.authority.com?token=ey… & [publickey] In this case the token is received with the link, and the public key must be replaced by the key of the new generated key pair of the certificate container in the wallet app. The deep link can be delivered by SMS, Email or by resenting another 2D Code for scan.		Schritt	Eingabe/Daten	Erwartete Ergebnisse
			1	Get a deep link by sms	dgc://example.authority.com?token=ey… & [publickey]	open deep link from sms
			2	WalletApp should start		WalletApp is opening and request biometric data
TXR-2200			3	scan correct biometric data		WalletApp starts the import of the certificate
	INT_WalletApp_use_deep_link_by_email_to_import_certificate	Optionally you can use a deep link instead of a 2D Code to initiate the certificate import in the wallet app. The deep link can look like: dgc://example.authority.com?token=ey… & [publickey] In this case the token is received with the link, and the public key must be replaced by the key of the new generated key pair of the certificate container in the wallet app. The deep link can be delivered by SMS, Email or by resenting another 2D Code for scan.		Schritt	Eingabe/Daten	Erwartete Ergebnisse
			1	Get a deep link by email	dgc://example.authority.com?token=ey… & [publickey]	open deep link from email
			2	WalletApp should start		WalletApp is opening and request biometric data
TXR-2201			3	scan correct biometric data		WalletApp starts the import of the certificate
		Optionally you can use a deep link instead of a 2D Code to initiate the certificate import in the wallet app. The deep link can look like:		Schritt	Eingabe/Daten	Erwartete Ergebnisse

	INT_WalletApp_negativ_use_fake_deep_link_by_SMS_to_import_certificate	dgc://example.authority.com?token=ey… & [publickey] In this case the token is received with the link, and the public key must be replaced by the key of the new generated key pair of the certificate container in the wallet app. The deep link can be delivered by SMS, Email or by resenting another 2D Code for scan.	1	Get a fake deep link by sms	dgc://example.authority.com?token=ey… & [publickey]	open deep link from sms
			2	WalletApp should start		WalletApp is opening and request biometric data
TXR-2202			3	scan correct biometric data		WalletApp shows an error with invalid link
	INT_WalletApp_negativ_use_fake_deep_link_by_email_to_import_certificate	Optionally you can use a deep link instead of a 2D Code to initiate the certificate import in the wallet app. The deep link can look like: dgc://example.authority.com?token=ey… & [publickey] In this case the token is received with the link, and the public key must be replaced by the key of the new generated key pair of the certificate container in the wallet app. The deep link can be delivered by SMS, Email or by resenting another 2D Code for scan.		Schritt	Eingabe/Daten	Erwartete Ergebnisse
			1	Get a fake deep link by email	dgc://example.authority.com?token=ey… & [publickey]	open deep link from email
			2	WalletApp should start		WalletApp is opening and request biometric data
TXR-2203			3	scan correct biometric data		WalletApp shows an error with invalid link
	INT_WalletApp_start_WalletApp_with_wrong_PIN	If the citizen has no biometric data on his mobile device it shoul be possible to start the device by PIN		Schritt	Eingabe/Daten	Erwartete Ergebnisse
			1	start the WalletApp on mobile device		biometric data are requested
			2	user push cancel		a user PIN is requested
TXR-2205			3	insert the wrong PIN		WalletApp shows an error