



## **Bakalářská práce**

# **Zavedení automatizace do procesu testování a uvolnění digitálních produktů**

*Studijní program:*

B0688P140002 Informační management

*Autor práce:*

**Vojtěch Haber**

*Vedoucí práce:*

Ing. Dana Nejedlová, Ph.D.

Katedra informatiky

Liberec 2024





## Zadání bakalářské práce

# Zavedení automatizace do procesu testování a uvolnění digitálních produktů

<i>Jméno a příjmení:</i>	<b>Vojtěch Haber</b>
<i>Osobní číslo:</i>	E21000539
<i>Studijní program:</i>	B0688P140002 Informační management
<i>Zadávací katedra:</i>	Katedra informatiky
<i>Akademický rok:</i>	2023/2024

### Zásady pro vypracování:

1. Úvod do aplikací a digitálních produktů.
2. Automatizace testování softwaru a jiné inovativní postupy.
3. Digitální produkty a procesy v konkrétní firmě.
4. Automatizace testování vybraného digitálního produktu.
5. Přínosy automatizace ve vývoji digitálních produktů pro podnikové procesy a ekonomiku.

*Rozsah grafických prací:*

*Rozsah pracovní zprávy:*

*Forma zpracování práce:*

*Jazyk práce:*

30 normostran

tištěná/elektronická

čeština

### **Seznam odborné literatury:**

- AXELROD, Arnon, 2018. *Complete Guide to Test Automation: Techniques, Practices, and Patterns for Building and Maintaining Effective Software Projects*. Berkeley, CA, USA: Apress. ISBN 978-1-4842-3831-8.
  - BUREŠ, Miroslav, Miroslav RENDA, Michal DOLEŽEL a kolektiv, 2016. *Efektivní testování softwaru: Klíčové otázky pro efektivitu testovacího procesu*. Profes!onal. Praha: Grada. ISBN 978-80-247-5594-6.
  - DŘÍMALKÁ, Filip, 2020. *HOT: jak uspět v digitálním světě. Pod povrchem*. Brno: Jan Melvil Publishing. ISBN 978-80-7555-101-6.
  - ROUDENSKÝ, Petr, 2018. *Kvalita softwaru: Teorie a praxe*. 2. upravené a rozšířené vydání. Prostějov: Computer Media. ISBN 978-80-7402-322-4.
  - VAN ROSSUM, Guido, 2015. The Early Years of Python. online. *Computer*, vol. 48, no. 2, s. 7–9. e-ISSN 1558-0814, ISSN 0018-9162. Dostupné z: <https://doi.org/10.1109/MC.2015.45>.
- Konzultant: Ing. Klauďie Olšáková - Projektový manažer konektivity, Škoda Auto a.s.

*Vedoucí práce:*

Ing. Dana Nejedlová, Ph.D.

Katedra informatiky

*Datum zadání práce:*

1. listopadu 2023

*Předpokládaný termín odevzdání:* 31. srpna 2025

L.S.

doc. Ing. Aleš Kocourek, Ph.D.  
děkan

Mgr. Tereza Semerádová, Ph.D.  
garant studijního programu

V Liberci dne 1. listopadu 2023

## Prohlášení

Prohlašuji, že svou bakalářskou práci jsem vypracoval samostatně jako původní dílo s použitím uvedené literatury a na základě konzultací s vedoucím mé bakalářské práce a konzultantem.

Jsem si vědom toho, že na mou bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci nezasahuje do mých autorských práv užitím mé bakalářské práce pro vnitřní potřebu Technické univerzity v Liberci.

Užiji-li bakalářskou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti Technickou univerzitu v Liberci; v tomto případě má Technická univerzita v Liberci právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Současně čestně prohlašuji, že text elektronické podoby práce vložený do IS/STAG se shoduje s textem tištěné podoby práce.

Beru na vědomí, že má bakalářská práce bude zveřejněna Technickou univerzitou v Liberci v souladu s § 47b zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů.

Jsem si vědom následků, které podle zákona o vysokých školách mohou vyplývat z porušení tohoto prohlášení.



# **Zavedení automatizace do procesu testování a uvolnění digitálních produktů**

## **Anotace**

Tato bakalářská práce se zabývá přínosy automatizace testování ve vývoji digitálních produktů a jejich dopady na podnikové procesy a ekonomiku. Práce je rozdělena do několika kapitol, z nichž každá pokrývá specifické aspekty tohoto tématu. Práce poskytuje základní přehled o rozdělení digitálních produktů, jejich historii a dopad digitalizace na automotive průmysl. Část práce se věnuje moderním technikám a nástrojům používaným při testování softwaru s důrazem na automatizaci. Dále se práce zabývá digitálními produkty a testovacími postupy pro zajištění jejich kvality a pokračuje popisem kroků implementace automatizace testování. V závěru práce shrnuje hlavní ekonomické a provozní výhody, které automatizace testování přináší.

## **Klíčová slova**

Digitální produkty, MyŠkoda, Automatizované testování, Automotive

# **Introducing Automation into the Process of Testing and Release of Digital Products**

## **Annotation**

This bachelor's thesis deals with the benefits of automated testing in the development of digital products and their impacts on business processes and the economy. The thesis is divided into several chapters, each covering specific aspects of this topic. It provides a basic overview of the categorization of digital products, their history, and the impact of digitalization on the automotive industry. A part of the thesis focuses on modern techniques and tools used in software testing with the emphasis on automation. Furthermore, the thesis addresses digital products and the testing procedures used to ensure their quality, followed by a description of the steps for implementing test automation. Lastly, the thesis summarizes the main economic and operational advantages that test automation brings in.

## **Key Words**

Digital products, MyŠkoda, Automated testing, Automotive



# Obsah

<b>Seznam ilustrací (obrázků) .....</b>	<b>11</b>
<b>Seznam použitých zkratk, značek a symbolů.....</b>	<b>12</b>
<b>Úvod .....</b>	<b>13</b>
<b>1 Úvod do aplikací a digitálních produktů .....</b>	<b>15</b>
1.1 Charakteristika a výhody digitálních produktů.....	15
1.2 Historie digitálních produktů .....	17
1.3 Vliv digitalizace na automotive průmysl.....	19
<b>2 Automatizace testování software a jiné inovativní postupy .....</b>	<b>22</b>
2.1 Rozdíly mezi manuálním a automatickým testováním a kroky pro implementaci automatizovaného testování.....	22
2.2 Jiné inovativní postupy vývoje software.....	23
2.3 Porovnávání jazyků Python a Java pro automatizaci .....	25
2.3.1 Python .....	25
2.3.2 Java.....	26
2.3.3 Porovnání jazyka Python a Java .....	26
<b>3 Digitální produkty a procesy ve firmě Škoda Auto a.s. ....</b>	<b>27</b>
3.1 Obecné představení produktů .....	27
3.1.1 Infotainment Online.....	27
3.1.2 Care Connect.....	28
3.1.3 Emergency Call a Other Services.....	30
3.1.4 Mobilní aplikace MyŠkoda.....	31
3.1.5 WebApps.....	31
3.2 Představení aplikace MyŠkoda.....	32
3.3 Popis procesu manuálního testování aplikace MyŠkoda .....	35
<b>4 Automatizace testování digitálního produktu aplikace MyŠkoda.....</b>	<b>38</b>
4.1 Analýza pro vývoj a popis nového testovacího scénáře .....	38
4.2 Vývoj předpřipraveného testu .....	46
<b>5 Přínosy automatizace ve vývoji digitálních produktů pro podnikové procesy a ekonomiku.....</b>	<b>51</b>
5.1 Přínosy automatizovaného testování pro kvalitnější produkt a procesy firmy.....	51
5.2 Dopad automatizovaného testování na ekonomiku .....	52

<b>Závěr .....</b>	<b>53</b>
<b>Seznam použité literatury .....</b>	<b>54</b>

## Seznam ilustrací (obrázků)

Obrázek 1: Přehled služeb Infotainment Online .....	28
Obrázek 2: Přehled služeb Care Connect: Proactive Service.....	28
Obrázek 3: Přehled služeb Care Connect: Remote Access .....	29
Obrázek 4: Functions on Demand.....	30
Obrázek 5: Other Services .....	31
Obrázek 6: Sekce aplikace MyŠkoda .....	33
Obrázek 7: Přehled funkcí MyŠkoda.....	34
Obrázek 8: Timeline sedmičkových verzí v roce 2024.....	35
Obrázek 9: Sekce car details a vygenerovaný certifikát .....	40
Obrázek 10: Ukázka prvního kroku testovacího scénáře.....	42
Obrázek 11: Přihlašování do MyŠkoda aplikace .....	43
Obrázek 12: Tlačítko inspect a car details.....	44
Obrázek 13: Kopírování VIN kódu a swipe .....	44
Obrázek 14: Text a tlačítko pro stažení a otevření digitálního certifikátu .....	45
Obrázek 15: Vygenerovaný digitální certifikát.....	46
Obrázek 16: Kód seznamu elementů.....	47
Obrázek 17: Importování funkcí.....	48
Obrázek 18: Ukázka hlavního kódu .....	50

## Seznam použitých zkratk, značek a symbolů

AI	Umělá inteligence ( <i>Artificial intelligence</i> )
CD	Kontinuální nasazení ( <i>Continuous Deployment</i> )
CI	Kontinuální integrace ( <i>Continuous Integration</i> )
CW	Kalendářní týden ( <i>Calendar Week</i> )
GDPR	Obecné nařízení o ochraně osobních údajů ( <i>General Data Protection Regulation</i> )
IoT	Internet věcí ( <i>Internet of Things</i> )
POI	Body zájmu ( <i>Points of Interest</i> )
RC	Kandidát pro uvolnění ( <i>Release Candidate</i> )
SaaS	Software jako služba ( <i>Software as a Service</i> )
SW	Software

# Úvod

V současné době, kdy digitální technologie pronikají do každodenního života i podnikatelského prostředí, je nezbytné efektivně využívat digitální produkty a inovativní postupy k dosažení úspěchu. Digitální produkty a aplikace jsou faktorem konkurenceschopnosti na globalizovaném trhu. V této situaci je důležitá nejen rychlost vývoje, ale také kvalita a spolehlivost digitálních produktů, což vede k rostoucímu zájmu o inovativní postupy v oblasti testování softwaru. Automatizace testování softwaru a další inovativní postupy jsou středem zájmu vývojářů, protože přinášejí efektivitu, rychlost a spolehlivost do vývoje digitálních produktů. Tato bakalářská práce je zaměřena na zkoumání těchto technologií a jejich dopad na vývoj digitálních produktů.

Cílem této bakalářské práce je provést komplexní analýzu a popis vlivu automatizace testování softwaru na vývoj digitálních produktů v konkrétním podnikovém prostředí. Dílčí cíle zahrnují analýzu a popis výhod a výzev spojených s implementací automatizace testování ve vývoji digitálních produktů, vyhodnocení efektivity testování a přínosy automatizace testování prostřednictvím analýzy dat získaných během implementace, implementování automatizace testování na vybraném digitálním produktu ve spolupráci s týmem testerů oddělení Smart kvality ve firmě Škoda Auto a.s. a zkoumání, jak automatizace testování ovlivňuje kvalitu a spolehlivost digitálních produktů.

Pro dosažení cílů této bakalářské práce budou využity metody, jako je literární rešerše, která poskytne přehled o současných poznacích v oblasti automatizace testování softwaru, dále analýza, která se zaměří na vyhodnocení efektivity a přínosů automatizace, a případová studie zaměřená na implementaci automatizace testování na vybraném digitálním produktu ve spolupráci s týmem testerů.

Tímto způsobem se práce snaží přispět k hlubšímu porozumění významu automatizace testování ve vývoji digitálních produktů a poskytnout praktický pohled na to, jak tato inovativní metoda může ovlivnit podnikové prostředí a ekonomiku.

Společnost Škoda Auto a.s. byla vybrána pro bakalářskou práci, protože je to jedna z předních a největších firem v České republice. Z důvodu svého významného postavení na trhu, poskytuje možnost získat hluboké a relevantní vhledy do procesů vývoje digitálních produktů.

Při přípravě textu byl využit program ChatGPT-4o pro 1 a 2 kapitolu.

# 1 Úvod do aplikací a digitálních produktů

Digitální produkty a technologie jsou v dnešní době všudypřítomné a ovlivňují všechny aspekty našeho života. Od komunikace a zábavy po práci a vzdělávání, digitální produkty mění způsob, jakým interagujeme se světem. Tato kapitola se zaměřuje na tři hlavní oblasti: charakteristika a výhody digitálních produktů, historie digitálních produktů a vliv digitalizace na automotive průmysl.

## 1.1 Charakteristika a výhody digitálních produktů

Digitální produkty jsou produkty, které ke své funkci využívají výpočetní techniku. Tato technologie umožňuje digitálním produktům poskytovat širokou škálu funkcí, které by nebyly možné u tradičních produktů.

Mezi digitálními produkty existuje 5 různých typů. Prvním typem digitálního produktu jsou softwarové aplikace. Tato kategorie zahrnuje širokou škálu aplikací, od nástrojů produktivity, jako jsou textové procesory a návrhářský software, až po aplikace pro zábavu, jako jsou hry a streamovací služby. Druhým typem digitálního produktu je digitální obsah. Digitální obsah zahrnuje různé formy včetně elektronických knih, hudby, videí a podcastů. Uživatelé mohou k tomuto obsahu přistupovat a využívat jej na digitálních (online) platformách. Dalším typem jsou online platformy. Sociální média, webové stránky elektronického obchodu, online tržiště a úložiště jsou příklady digitálních platform, které usnadňují interakci uživatelů, finanční transakce a sdílení obsahu. Čtvrtým produktem je SaaS (Software as a Service, v češtině software jako služba). Produkty SaaS jsou softwarová řešení dostupná přes internet, takže uživatelé nemusí instalovat, udržovat a aktualizovat software lokálně, jako je to v případě softwarových aplikací uvedených jako první typ digitálního produktu. Příklady zahrnují cloudové nástroje pro spolupráci a systémy řízení vztahů se zákazníky. Posledním typem digitálních produktů jsou digitální služby. Do této kategorie spadají služby poskytované přes internet, jako je online vzdělávání, telemedicína a služby digitálního marketingu. Je nutno dodat, že většina digitálních produktů je zpoplatněna a nejsou volně přístupné. (Productfolio, 2024)

Konkrétním příkladem digitálního produktu ve formě softwarové aplikace je Microsoft Office, který zahrnuje aplikace jako Word, Excel a PowerPoint. Tento produkt je dostupný jak ve formě instalovatelných aplikací, tak ve formě cloudové služby Office 365, čímž spadá Microsoft Office i do kategorie SaaS. Uživatelé mají přístup ke svým dokumentům odkudkoli a mohou je snadno sdílet a spolupracovat s ostatními, což znamená, že využívají také online platformu. Příkladem druhé kategorie je digitální obsah dostupný na Amazon Kindle, zařízení a platformě pro čtení elektronických knih. Amazon Kindle umožňuje uživatelům přístup ke statisícům knih, novin a časopisů, které si mohou okamžitě stáhnout a číst. Spotify je příkladem digitálního produktu ve formě online platformy v oblasti hudby. Tato streamovací služba poskytuje přístup k milionům skladeb a podcastů. Uživatelé mohou vytvářet vlastní playlisty, objevovat novou hudbu a sdílet své oblíbené skladby s přáteli. Díky modelu předplatného Spotify nabízí neomezený přístup k hudbě bez reklam, což zvyšuje komfort užívání. Příkladem digitální služby je Coursera. Coursera je platforma pro online vzdělávání, která nabízí kurzy z prestižních univerzit a institucí po celém světě. Uživatelé mohou studovat širokou škálu předmětů od programování po marketing a získávat certifikáty, které mohou zlepšit jejich kariérní vyhlídky. Díky možnosti studovat kdykoli a odkudkoli Coursera zpřístupňuje vzdělání širšímu okruhu lidí.

Digitální produkty mají v dnešní době mnoho výhod.

Za prvé, digitalizace umožňuje firmám zlepšit efektivitu a spolehlivost jejich operací, což vede k vyšší konkurenceschopnosti a inovativnosti. Technologie jako umělá inteligence, internet věcí, strojové učení a blockchain umožňují optimalizovat celé pracovní procesy a dodavatelské řetězce. (Berawi et al., 2020)

Další výhodou je vysoká flexibilita a multifunkčnost digitálních produktů. Tyto produkty, zahrnující software, e-knihy, digitální hudbu a filmy, mohou být distribuovány přes internet s minimálními transakčními náklady, což usnadňuje jejich globální dostupnost a využitelnost. (Pernul, 2003)

Digitální produkty také umožňují lepší interakci a propojení mezi uživateli a systémy. To vede ke zvýšené personalizaci a možnosti integrace zákaznických požadavků



přímo do vývoje produktů a služeb. Tento přístup podporuje tvorbu přidané hodnoty a posiluje vztahy mezi firmami a jejich zákazníky. (Holmquist, 2012)

Digitální produkty ve formě platform umožňují flexibilní nastavení ceny pro uživatele v závislosti na objemu, ve kterém si přeje službu využívat. (Alhamad et al., 2011)

Konečně, digitální produkty přinášejí výhody i v oblasti vzdělávání, kde umožňují flexibilní a interaktivní učení. Tento způsob výuky se ukázal jako obzvláště efektivní během pandemie, kdy bylo nutné přejít na distanční vzdělávání. Učitelé a studenti oceňují výhody, které digitální produkty přinášejí, jako je vyšší efektivita a možnost individuálního přístupu k materiálům. (Puchkova et al., 2021)

Závěrem lze uvést, že digitální produkty přinášejí širokou škálu výhod, které zlepšují efektivitu, interaktivitu, dostupnost a personalizaci v různých oblastech lidské činnosti.

## **1.2 Historie digitálních produktů**

Historie digitálních produktů sahá až do období, kdy byly vyvinuty první elektronické digitální počítače. Ve 40. letech 20. století byly první digitální počítače obrovské, nespolehlivé a velmi nákladné. Například ENIAC, jeden z prvních elektronických digitálních počítačů, zabíral celou místnost a vyžadoval rozsáhlou údržbu. Nicméně postupem času se tyto počítače zmenšovaly, zlepšovala se jejich spolehlivost a snižovaly se náklady (Haigh, 2019)

V 60. letech 20. století došlo k významnému přechodu počítačů od specializovaných vědeckých nástrojů k běžným komerčním produktům. Společnosti jako IBM začaly vyrábět počítače, které byly dostupné široké veřejnosti a firmám. Tento přechod byl umožněn díky technologickým pokrokům, jako je vývoj polovodičových čipů, které umožnily zmenšení velikosti a snížení nákladů na výrobu počítačů.

V 70. letech 20. století se začaly objevovat první osobní počítače. Společnosti jako Apple a Microsoft začaly vyrábět počítače, které byly určeny pro domácí a osobní použití. Tento vývoj byl umožněn díky pokrokům v mikroprocesorové technologii, která umožnila výrobu malých, ale výkonných počítačů. Osobní počítače se rychle staly populárními a začaly měnit způsob, jakým lidé pracovali a komunikovali (Ceruzzi, 2012)

V 80. a 90. letech 20. století došlo k dalšímu rozvoji digitálních produktů. Vývoj internetu a vznik World Wide Webu umožnily globální propojení lidí a počítačů. To vedlo k rozvoji nových digitálních produktů a služeb, jako jsou e-mail, online obchody a sociální sítě. Digitální produkty se staly nedílnou součástí každodenního života a začaly měnit způsob, jakým lidé komunikovali, pracovali a trávili volný čas. (Ceruzzi, 2012)

V současnosti digitální produkty zahrnují širokou škálu produktů. Tyto produkty využívají pokročilé technologie, jako je umělá inteligence a strojové učení, k poskytování personalizovaných a inteligentních služeb. Digitální produkty se staly klíčovou součástí moderní společnosti a jejich význam bude i nadále růst s pokračujícím technologickým pokrokem (Holmquist, 2012)

Microsoft je jedním z nejvýznamnějších průkopníků v oblasti digitálních produktů. Jejich operační systém Windows a softwarový balík Microsoft Office se staly standardem v osobních a podnikových počítačích. Microsoft také hrál klíčovou roli v rozvoji cloudových služeb s platformou Azure, která poskytuje širokou škálu cloudových řešení pro firmy.

Apple je další významný průkopník, který změnil trh s digitálními produkty svými inovativními zařízeními a službami. iPhone, iPad a MacBook se staly ikonickými produkty, které definovaly moderní digitální životní styl. Apple také vytvořil úspěšný ekosystém aplikací a služeb, jako je App Store a iCloud, který umožňuje uživatelům bezproblémově integrovat a synchronizovat své digitální zážitky.

Google je dalším klíčovým hráčem, který změnil svět digitálních produktů. Jejich vyhledávač, Gmail, Google Maps a operační systém Android jsou jen některé

z produktů, které se staly nepostradatelnými v každodenním životě milionů lidí. Google také investoval do pokročilých technologií, jako jsou umělá inteligence a autonomní vozidla, které mají potenciál dále transformovat digitální svět.

Stojí také za zmínku jeden z klíčových regulačních milníků, a to směrnice Evropské unie o ochraně osobních údajů (GDPR), která vstoupila v platnost v roce 2018. Směrnice GDPR zavedla přísné požadavky na ochranu osobních údajů uživatelů a uložila firmám povinnost informovat uživatele o tom, jak jsou jejich data sbírána a používána. Tato směrnice měla významný dopad na digitální produkty a služby, které musely přizpůsobit své postupy a technologie, aby byly v souladu s novými předpisy.

### **1.3 Vliv digitalizace na automotive průmysl**

Digitalizace výrazně mění automobilový průmysl a boří tradiční obchodní modely. Digitalizace zahrnuje širokou škálu technologií a služeb, které mají zásadní dopad na celý sektor, od výroby až po prodej a poprodejní služby. Tento text se zaměřuje na několik hlavních aspektů digitalizace a jejich vliv na automobilový průmysl.

Digitalizace mění tradiční obchodní modely v automobilovém průmyslu. Společnosti jsou nuceny přijmout nové strategie a přizpůsobit se digitálním technologiím, aby zůstaly konkurenceschopné. Digitalizace umožňuje vznik nových produktů, jako jsou autonomní a připojená vozidla, a digitálních služeb, jako jsou mobility services. Tento přechod zahrnuje personalizaci služeb a řešení zaměřených na individuální interakce se zákazníky, což zahrnuje například omnichannel předprodejní engagement a personalizované pojištění. (Deryabina & Trubnikova, 2021) (Verevka et al., 2019).

Digitalizace výrazně zlepšuje výrobní procesy v automobilovém průmyslu. Průmysl 4.0, který kombinuje automatizaci, výměnu dat a digitální technologie, vede k efektivnějším a flexibilnějším výrobním prostředím. Integrace internetu věcí (IoT) umožňuje vozidlům shromažďovat, zpracovávat a komunikovat obrovské množství dat, což vede k lepší bezpečnosti, optimalizaci spotřeby paliva a prediktivní údržbě. Výrobní procesy jsou stále více propojené a kognitivní, což umožňuje lepší adaptaci

na potřeby zákazníků a změny ve výrobních podmínkách. (Armengaud et al., 2017) (Monye et al., 2023)

Digitalizace také mění strukturu zaměstnanosti v automobilovém průmyslu. Automatizace a digitální technologie ovlivňují typy pracovních míst a požadované dovednosti. Například potřeba nových dovedností v oblasti softwarového vývoje, správy dat a kybernetické bezpečnosti roste, což vyžaduje upskilling a reskilling zaměstnanců. Tento posun přináší nové výzvy, ale také příležitosti pro vytváření nových pracovních míst a zvyšování produktivity. (Felser & Wynn, 2023)

Digitalizace výrazně ovlivňuje zákaznické zkušenosti a způsob, jakým jsou automobily prodávány a servisovány. Díky digitálním technologiím mají zákazníci přístup k lepším službám, což zvyšuje jejich spokojenost. Digitální služby, jako jsou online konfigurace vozidel a virtuální prohlídky, mění způsob, jakým zákazníci interagují s automobilovými značkami. Tradiční výrobci automobilů musí investovat do digitálních platforem a služeb, aby si udrželi konkurenceschopnost vůči novým hráčům, jako jsou Apple a Google, kteří vstupují na automobilový trh. (Tian et al., 2016) (Llopis-Albert et al., 2020)

Digitalizace podporuje také ekologické inovace, jako jsou elektromobily a udržitelné technologie. Přechod k elektromobilitě je podporován environmentálními politikami a směrnicemi zaměřenými na snížení emisí a využití obnovitelných zdrojů energie. Tento posun má pozitivní dopad na životní prostředí a vytváří nové obchodní příležitosti. (Llopis-Albert et al., 2020).

Tesla, jako průkopnická firma v oblasti elektrických vozidel, má zásadní vliv na automobilový průmysl. Její inovativní obchodní strategie a technologické inovace změnily pravidla hry a vedly k výrazným posunům v celém odvětví. Tesla se stala lídrem na trhu s elektrickými vozidly díky své silné značce, technologickým průlomům a schopnosti rychle přizpůsobit své produkty požadavkům trhu. Společnost si udržuje konkurenční výhodu díky své schopnosti inovovat a snižovat náklady. Tesla změnila tradiční pojetí výroby automobilů tím, že se zaměřila na software a elektronickou architekturu vozidel. Tento přístup přinesl přehodnocení hodnotového systému automobilového průmyslu a otevřel dveře novým obchodním modelům. Tesla tak vytvořila tzv. "Catfish Effect" (ve smyslu „Vypuštění sumce

mezi menší ryby je donutí k větší aktivitě."), který podnítil další společnosti k inovacím a změně přístupu. Tesla má významný vliv na automobilový průmysl, a to díky svým inovativním obchodním strategiím, technologickým inovacím a zaměření na udržitelnost. Její přístup změnil pravidla hry a inspiroval ostatní společnosti k přijetí ekologičtějších a technologicky vyspělejších přístupů. (Bao, 2022) (Wu, 2023)

Digitalizace má hluboký a široký dopad na automobilový průmysl, transformující obchodní modely, výrobní procesy, trh práce, zákaznické zkušenosti a ekologické inovace. Společnosti, které se dokáží přizpůsobit a využít výhod digitalizace, budou schopny dosáhnout vyšší konkurenceschopnosti a spokojenosti zákazníků.

## **2 Automatizace testování software a jiné inovativní postupy**

V dnešní době existuje mnoho přístupů k testování software a zlepšování jeho kvality. V této kapitole budou znázorněny rozdíly mezi automatizovaným a manuálním testováním. Dále budou popsány jednoduché kroky při implementaci automatizovaného testování, další inovativní postupy pro testování software a nakonec budou porovnány programovací jazyky využívané pro testování.

### **2.1 Rozdíly mezi manuálním a automatickým testováním a kroky pro implementaci automatizovaného testování**

Jak píše Petr Roudenský v knize Kvalita softwaru *“Infrastruktura současné moderní společnosti je řízena prostřednictvím software, který představuje klíčový prvek a jehož role a s ní i naše závislost na jeho funkci stále roste.”* (Roudenský, 2018, s.16) S tímto roste i význam a potřeba pro bezpečnost, spolehlivost a kvalitu software. Automatizace nabízí možnost zlepšení software oproti klasickému manuálnímu testování. (Roudenský, 2018)

Je zde také důležité zmínit pojem kvalitativní testování. Kvalitativní testování je zaměřené na pohled zákazníka a snaží se porozumět jejich zkušenostem, potřebám a očekáváním. Používá metody jako uživatelské rozhovory, testování použitelnosti a fokusní skupiny, které umožňují získat návrhy na zlepšení přímo od zákazníků. Tento přístup pomáhá organizacím vytvářet produkty a služby, které lépe odpovídají potřebám uživatelů a zlepšují jejich celkový zážitek.

Automatické testování má oproti manuálnímu několik výhod. Je rychlejší a efektivnější, poskytuje konzistentní výsledky a je snadno škálovatelné, což umožňuje vyšší pokrytí testovacích případů. Dále umožňuje včasné odhalení defektů a je nezbytné pro kontinuální integraci a doručování. Na druhou stranu, automatické testování vyžaduje vysoké počáteční náklady a údržbu, je citlivé na změny v aplikaci a není vhodné pro průzkumné testování.

Manuální testování je flexibilní, levnější na počátku a umožňuje testerům využít jejich intuici a zkušenosti. Nicméně, je časově náročné, méně efektivní pro opakující se testy, náchylné k lidským chybám a špatně škálovatelné pro velké testovací sady. (Axelrod, 2018)

Kniha "Complete Guide to Test Automation" od Arnona Axelroda popisuje několik klíčových kroků k úspěšné automatizaci testování. Nejprve je nutné navrhnout první testovací případ, identifikovat, co je třeba testovat, a vytvořit detailní plán. Poté se píše pseudokód a vytváří se minimální infrastruktura potřebná ke kompilaci testu. Následuje plná implementace testovacího případu a zajištění jeho správného běhu.

Tento proces se opakuje pro další testovací případy s důrazem na udržovatelnost kódu. Automatické testy se integrují do kontinuální integrace (CI), aby se testy spouštěly jako součást CI pipeline. Postupně se rozšiřuje testovací sada na více scénářů a různé typy testů.

Je důležité vyvinout strategie pro řešení selhání testů, vytváření jasných chybových zpráv a logů, což jsou data, kde jsou popsány přesné kroky testované aplikace a její chování. Logy jsou obvykle vytvářeny automatizačními nástroji a testovacími frameworky, často s podporou vývojářů. Nakonec je nutné pravidelně aktualizovat a udržovat testovací případy, aby odpovídaly změnám v aplikaci a zůstaly relevantní a přesné. Tyto kroky vytvářejí robustní rámec pro efektivní a udržitelné automatické testování. (Axelrod, 2018)

Jedním z nejpoužívanějších software pro evidenci chyb je nástroj jménem JIRA. JIRA je softwarový nástroj pro evidenci chyb a problémů při vývoji softwaru nebo řízení projektů, vyvíjený společností Atlassian. JIRA podporuje a usnadňuje proces řízení projektů a požadavků, nabízí flexibilní a uživatelské nástroje pro řízení a sledování pracovníků při výkonu plnění úkolů.

## **2.2 Jiné inovativní postupy vývoje software**

Vývoj softwaru se neustále vyvíjí a inovuje, přičemž moderní přístupy jako agilní metodologie, DevOps a využití umělé inteligence hrají klíčovou roli v tomto procesu.

Tyto přístupy přinášejí nové možnosti, jak zlepšit efektivitu, kvalitu a rychlost dodávek softwaru, což je zásadní pro konkurenční výhodu na dnešním dynamickém trhu.

Agilní metodologie představuje flexibilní a iterativní přístup k vývoji softwaru, který klade důraz na rychlé dodávky funkčních produktů a neustálou komunikaci se zákazníky. Tento přístup umožňuje týmům rychle reagovat na změny požadavků a minimalizovat rizika spojená s dlouhodobým plánováním. Agilní metodiky se ukázaly jako velmi efektivní zejména při vývoji IT inovací v knihovnách, kde flexibilita a iterativní cykly umožňují rychlou tvorbu a nasazení inovativních aplikací (Chang, 2010).

DevOps je metodologie, která propojuje vývoj (Development) a provoz (Operations) s cílem zlepšit spolupráci mezi těmito dvěma oblastmi a zajistit kontinuální dodávku a integraci softwaru. DevOps přístupy zahrnují automatizaci procesů, jako je Continuous Integration (CI) a Continuous Delivery (CD), což zvyšuje efektivitu a kvalitu vývoje softwaru. Zavedení DevOps metodologie v kombinaci s cloud computingem může přinést inovativní služby s nižšími náklady a vyšší kvalitou. Tento přístup nejen zkracuje dobu potřebnou k uvedení produktu na trh, ale také zlepšuje úspěšnost projektů jak v průmyslu, tak ve vzdělávacích institucích (Airaj, 2017).

Umělá inteligence (AI) nabízí nové možnosti pro automatizaci a optimalizaci různých aspektů vývoje softwaru, včetně testování, analýzy kódu a predikce výkonnosti. AI může například automatizovat generování testovacích případů a detekci anomálií, což zvyšuje efektivitu a kvalitu softwarových produktů. Studie (Goel et al., 2023) popisuje použití AI pro automatizované generování testovacích případů, detekci anomálií a prediktivní analýzu, což zvyšuje efektivitu testování.

Inovativní přístupy jako agilní metodologie, DevOps a umělá inteligence přinášejí významné zlepšení do vývoje softwaru. Tyto přístupy umožňují rychlejší a kvalitnější dodávky softwaru, lepší spolupráci mezi týmy a efektivnější využití technologických zdrojů. Implementace těchto metodik a technologií je klíčová pro úspěch v současném konkurenčním prostředí.



## 2.3 Porovnávání jazyků Python a Java pro automatizaci

Automatizace testování je klíčovým prvkem v moderním vývoji softwaru, kde volba správného programovacího jazyka může výrazně ovlivnit efektivitu a kvalitu testování. Python a Java jsou dva z nejpopulárnějších jazyků používaných pro tento účel. Níže jsou uvedeny hlavní výhody a nevýhody obou jazyků pro automatizaci testování.

### 2.3.1 Python

Python byl vytvořen Guidem van Rossumem za účelem zefektivnění vývoje nástrojů pro operační systém Amoeba, protože vývoj v jazyce C byl příliš pomalý. Van Rossum byl inspirován jazykem ABC, který byl jednoduchý na použití, ale nevhodný pro systémové programování. Proto chtěl vytvořit nový jazyk, který by kombinoval jednoduchost ABC s potřebnou funkcionalitou. Během dovolené v prosinci 1989 začal ve svém volném čase experimentovat a vytvořil jazyk Python, pojmenovaný po komediálním seriálu Monty Python's Flying Circus. (Van Rossum, 2015)

Python je proto známý svou jednoduchou a čitelnou syntaxí, což usnadňuje rychlé psaní a údržbu testovacích skriptů. Díky tomu je ideální pro začátečníky i zkušené programátory. Python má také rozsáhlou knihovnu pro automatizaci testování, včetně nástrojů jako Selenium, pytest a unittest, což zjednodušuje psaní a správu testovacích skriptů. Díky své jednoduchosti a velkému množství dostupných knihoven umožňuje Python rychlejší vývoj a prototypování automatizačních testů ve srovnání s jinými jazyky. (Bogdanchikov et al., 2013) (Garcia, 2008)

Na druhou stranu, Python je obecně pomalejší než Java, což může být limitující při testování aplikací, které vyžadují vysoký výkon nebo manipulaci s velkými objemy dat. Python může být také méně efektivní ve správě paměti ve srovnání s Java, což může vést k vyšší spotřebě paměti při testování velkých aplikací. (Agu et al., 2016)

### **2.3.2 Java**

Java je kompilovaný jazyk, který poskytuje vyšší výkon a efektivitu ve srovnání s interprety, jako je Python. To je výhodné pro testování aplikací, které vyžadují rychlé a efektivní provádění. Java také nabízí robustní podporu pro vícevláknové programování, což je užitečné při testování aplikací, které využívají paralelní zpracování. Java je známá svou stabilitou a robustností, což z ní činí ideální volbu pro dlouhodobé projekty a aplikace vyžadující vysokou spolehlivost. (Putranto et al., 2020) (Durelli, 2013)

Nicméně, Java má složitější a méně přehlednou syntaxi ve srovnání s Pythonem, což může ztěžovat rychlé psaní a údržbu testovacích skriptů. Psát kód v Java může být časově náročnější než v Pythonu, což může zpomalit vývoj a testování aplikací. (Agu et al., 2016) (Lo et al., 2015)

### **2.3.3 Porovnání jazyka Python a Java**

Závěrem lze říct, že Python a Java mají své specifické výhody a nevýhody při použití pro automatizaci testování. Python vyniká svou jednoduchostí, rychlostí vývoje a širokou podporou knihoven, což z něj činí ideální volbu pro rychlé prototypování a testování menších aplikací. Na druhou stranu, Java nabízí vyšší výkon, spolehlivost a podporu pro více vláknové programování, což je výhodné pro testování rozsáhlých a náročných aplikací.

## **3 Digitální produkty a procesy ve firmě Škoda Auto a.s.**

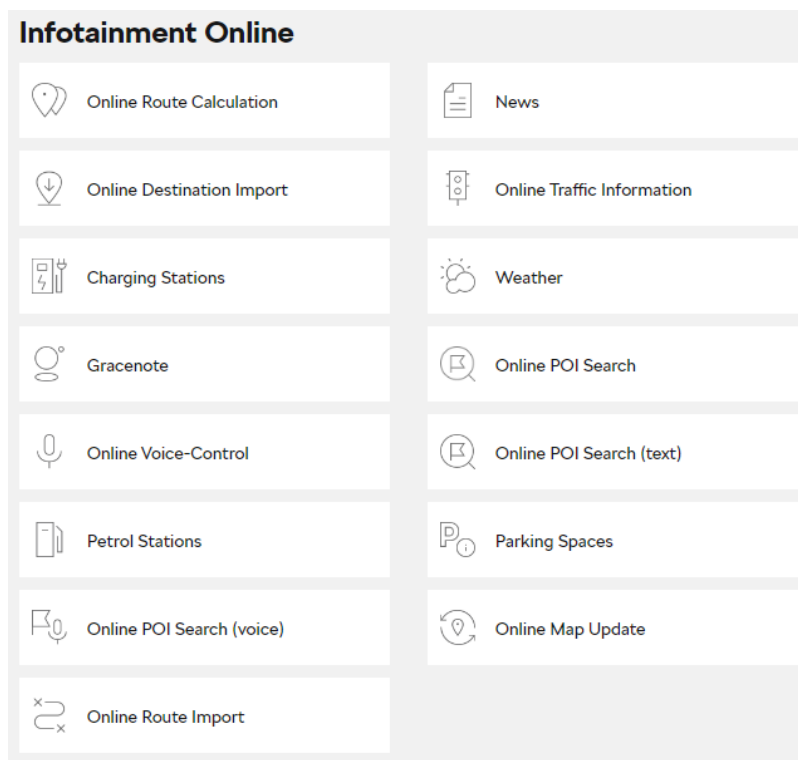
Tato kapitola se zaměřuje na zkoumání digitálních produktů a procesů ve firmě Škoda Auto a.s. Cílem této kapitoly je porozumět digitálním produktům této firmy a procesu manuálního testování jednoho z těchto produktů.

### **3.1 Obecné představení produktů**

Škoda Auto a.s. nabízí řadu digitálních produktů, které zvyšují komfort a bezpečnost jejich vozidel. Mezi hlavní digitální produkty patří Škoda Connect, což je komplexní sada online služeb rozdělených na Infotainment Online, Care Connect, který se dělí na Care Connect: Proactive Service, Care Connect: Remote Access a Care Connect: iV Remote Services. Dále sem patří služby Emergency Call a Other Services. Tyto služby jsou využívány skrze vůz nebo aplikace Škody Auto a.s.

#### **3.1.1 Infotainment Online**

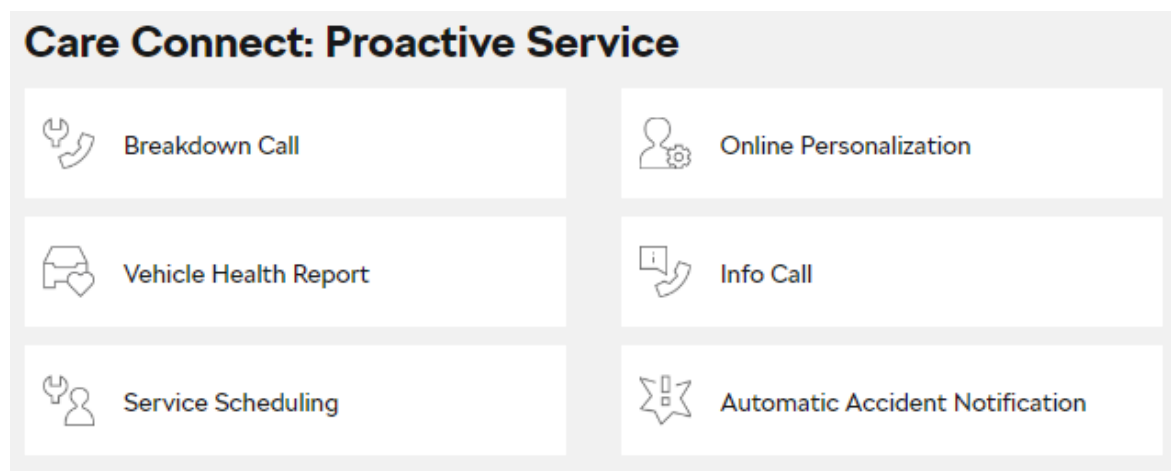
Pod Infotainment Online spadají například služby, které poskytují aktuální informace o dopravě, zprávy, informace o parkování, ceny paliva a další, umožňuje také online aktualizaci map a hlasové ovládání. Celkový přehled služeb v balíku Infotainment Online je možno vidět na obrázku 1.



Obrázek 1: Přehled služeb Infotainment Online  
 Zdroj: Škoda Auto a.s., 2023

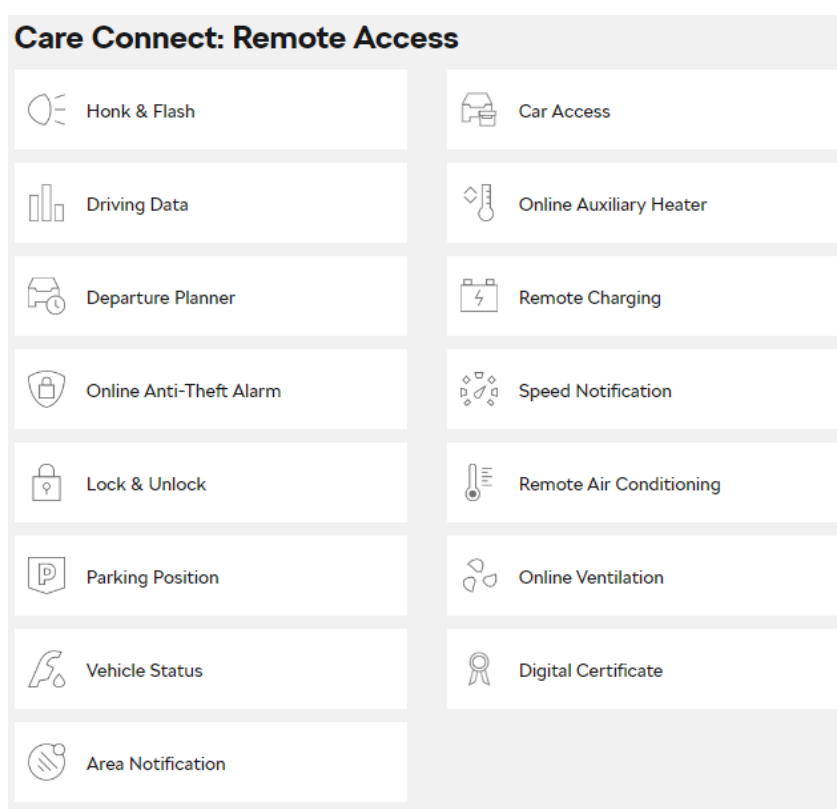
### 3.1.2 Care Connect

Care Connect: Proactive Service zahrnuje služby, které týkají technického stavu vozu. Může být zde kontrolováno „zdraví“ vozu, plánování navštívení servisu, automatické volání podpory při poruše auta a více. Jak je vidět na obrázku 2, jedná se o balík služeb s menším počtem funkcí než v balíku Infotainment Online.



Obrázek 2: Přehled služeb Care Connect: Proactive Service  
 Zdroj: Škoda Auto a.s., 2023

Jak již vyplývá z jeho názvu, další balíček služeb Care Connect: Remote Access umožňuje zákazníkům vzdálený přístup k autu. Tento balíček obsahuje nejrůznější služby pro získání informací o autě. Například funkce digitálního certifikátu na, kterou se zabývá kapitola 4 této práce, umožňuje vygenerování digitálního certifikátu s informacemi o autě, jako je výbava, najeté kilometry a tak dále. Tento certifikát poté usnadňuje práci při prodeji vozidla nebo v situacích, kde je potřeba získat přehled informací o autě. Díky službám z tohoto balíku může zákazník také ovládat auto na dálku. Zapínání vyhřívání nebo klimatizace, vzdálené nabíjení a také vzdálené odemykání a zamykání. Tento balík nabízí více služeb, které lze vidět na obrázku 3.



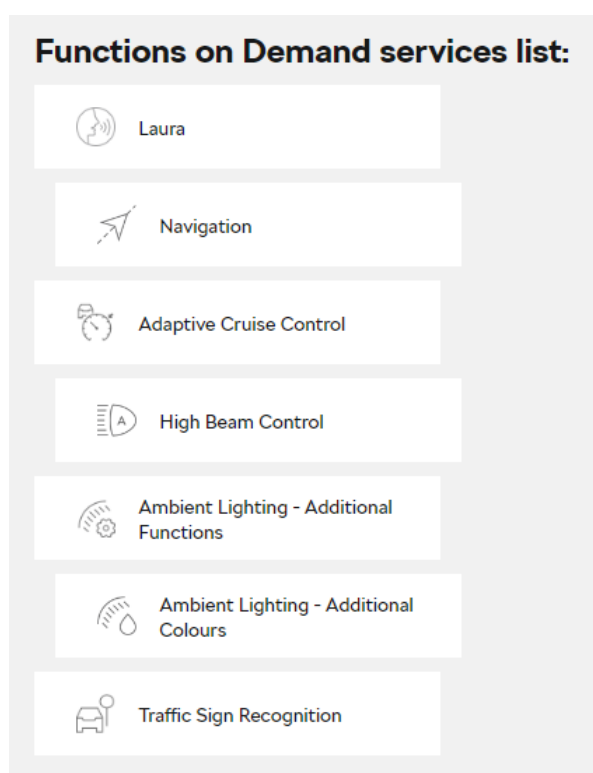
Obrázek 3: Přehled služeb Care Connect: Remote Access  
Zdroj: Škoda Auto a.s., 2023

Care Connect: iV Remote Services jsou služby pouze pro plně elektrický vůz Enyaq. Tento balík služeb nabízí stejné funkce jako výše představený balík Care Connect: Remote Access, ale nabízí navíc funkci Plug & Charge, která se používá k nabíjení na veřejných nabíječkách a umožňuje nabíjení bez verifikace, čímž umožňuje rychlejší nabíjení.

### 3.1.3 Emergency Call a Other Services

Mezi poslední balíčky služeb patří Emergency Call, který nabízí automatické volání pomoci při nehodě automobilu, a Other Services. (ostatní služby). Nejdůležitější funkcí v ostatních službách jsou Functions on Demand a SmartLink.

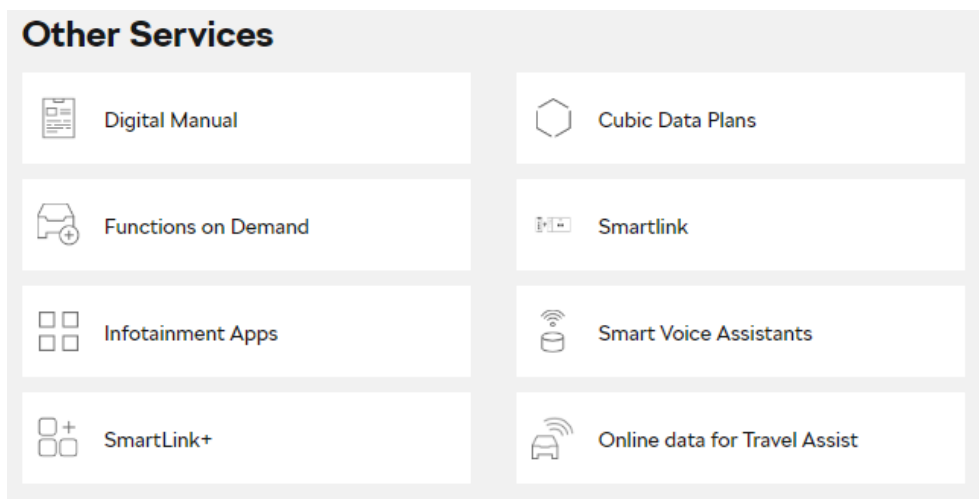
Functions on Demand je technologie pro jednoduché upgradování auta s novými funkcemi. Veškeré poplatky a aktivace jsou řešeny online. Mezi funkce Functions on Demand patří například navigace, Laura (AI asistent), kontrola světel nebo rozpoznávání značek při jízdě. Všechny funkce je možno vidět na obrázku 4.



Obrázek 4: Functions on Demand

Zdroj: Škoda Auto a.s., 2023

SmartLink slouží k propojení telefonu a auta pomocí kabelu. Poté je možné využívat telefon bez nutnosti jeho držení s využitím rádia nebo infotainmentu. SmartLink+ umožňuje sledování dat o autě v reálném čase pomocí aplikace MyŠkoda. Veškeré funkce balíku Other Services lze vidět na obrázku 5.



Obrázek 5: Other Services  
Zdroj: Škoda Auto a.s., 2023

### 3.1.4 Mobilní aplikace MyŠkoda

Pro používání věch výše představených funkcí a k lepšímu přehledu o autě představila Škoda Auto a.s. různé telefonní aplikace. Hlavní aplikací je MyŠkoda. Tato aplikace umožňuje využívání všech služeb od kontrolování nabití baterie, sledování stavu vozu nebo používání aplikace Powerpass k nabíjení. Powerpass byla aplikace pro nabíjení, ve které byly na mapě ukázány veškeré nabíjecí stanice, a bylo umožněno touto aplikací platit. Nyní je tato aplikace integrovaná do MyŠkoda, kde si stačí přidat kartu Powerpass a zákazník může platit pomocí aplikace MyŠkoda. Powerpass se používá pro funkci Plug & Charge. Další mobilní aplikací pro spouštění aplikací firmy Škoda Auto a.s je MyŠkoda Essentials. Tato aplikace je alternativou MyŠkoda a byla vytvořena pro starší generace vozů, které nebyly podporovány. Jednou z dalších aplikací je MyŠkoda iV. Pomocí této aplikace může zákazník porovnávat elektrické a spalovací motory a, zda se mu vyplatí využívat stávající auto, anebo koupit nové.

### 3.1.5 WebApps

Posledním digitálním produktem, který byl ve Škoda Auto a.s. dosud vyvinut, jsou WebApps. WebApps jsou standardní aplikace, jako například mobilní aplikace, které běží v autě vybaveném potřebným hardwarem. WebApps se skládají ze dvou částí, a to frontend, což je část, kterou zákazník vidí, a backend, což je část, kterou

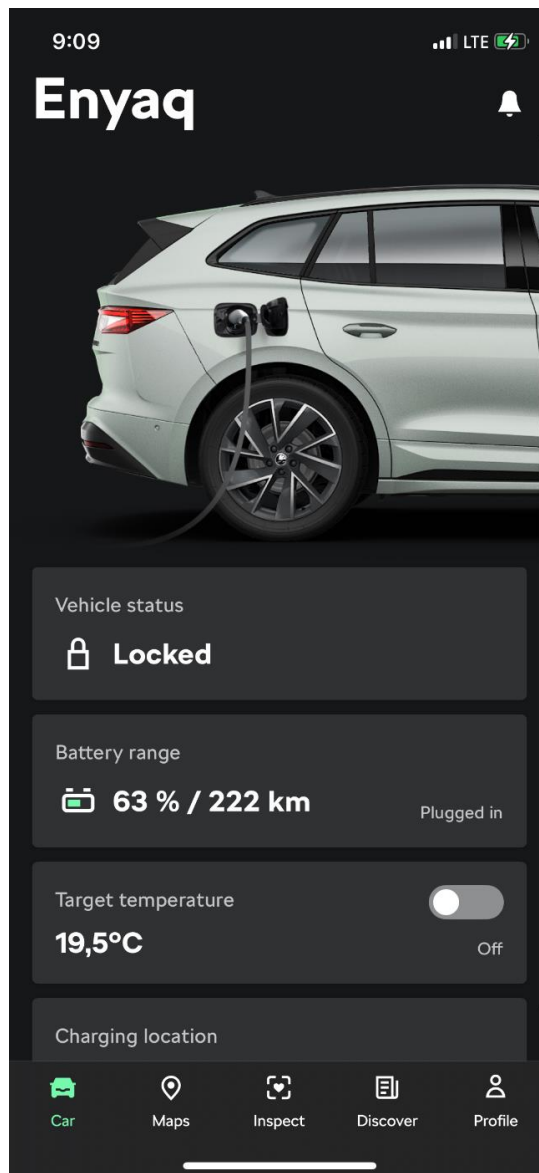
zákazník nevidí, a která běží na pozadí a využívá a ukládá nasbíraná data. Škoda Auto a.s. zatím nabízí osm WebApps: Powerpass Map, Weather, Calendar, Traffication, Offers, Pay to Fuel, Pay to Park a News. Powerpass Map nabízí přehled o volně přístupných nabíjecích stanicích, jejich hodnocení a zpětnou vazbu od zákazníků. Weather neboli počasí je aplikace, která dává předpověď počasí pro stávající a finální polohu. Calendar neboli kalendář dává možnost zákazníkovi vyřizovat emaily, zapisovat do kalendáře a účastnit se online konferencí v autě. Traffication monitoruje okolí auta a dává varovná oznámení. Offers nabízí nové produkty k zakoupení pro zákazníka. Pay to Fuel umožňuje zaplatit z auta při tankování. To stejné platí pro Pay to Park, co se týče parkování. News přeloženo jako novinky je aplikace pro získávání nejnovějších informací ze světa na základě preferencí zákazníka.

## 3.2 Představení aplikace MyŠkoda

MyŠkoda App je mobilní aplikace, která poskytuje užitečné funkce a služby pro majitele vozidel značky Škoda. Aplikace je navržena tak, aby umožnila majitelům vozidel snadný přístup k důležitým informacím a funkcí spojeným s jejich vozidlem, a to přímo z jejich chytrých telefonů.

MyŠkoda se skládá z pěti základních sekcí. Každá z těchto sekcí má na starosti různé funkce. Tyto sekce se nazývají *Car (Vůz)*, *Maps (Mapy)*, *Inspect (Prohlídka)*, *Discover (Discover)*, *Profile (Profil)*. Dole na obrázku 6 je možno vidět všech pět sekcí. Obrázek je v angličtině, protože celý projekt automatizovaného testování je prováděn v anglickém jazyce, proto pro zachování autentičnosti je obrázek ponechán v angličtině.





Obrázek 6: Sekce aplikace MyŠkoda  
Zdroj: vlastní snímek obrazovky aplikace MyŠkoda

Sekce *Car* je domovskou stránkou aplikace. Nachází se zde funkce pro ovládání auta na dálku. Tato sekce je rozdělena do 4 různých záložek. *Vehicle status*, *Battery range*, *Target Temperature* a *Charging locations*. Pomocí těchto záložek je zákazník schopen například zamykat auto na dálku, měnit teplotu klimatizace nebo zapnout nabíjení na dálku. V sekci *Maps* jsou veškeré funkce spojené s navigací, jako je vyhledávání destinací a jejich zaslání do auta, funkce *Pay to Fuel* a *Pay to Park*, ukládání oblíbených lokací nebo hledání POI (Points of interest – Bod zájmu). Všechny tyto funkce a některé další jsou součástí sekce *Maps*. Sekce *Inspect* slouží ke kontrole stavu vozidla a možného objednání do servisu. V sekci *Discover* jsou novinky ze světa. V neposlední řadě v sekci *Profile* se nachází nastavení aplikace,

právní dokumenty nebo nastavení profilu. Samozřejmě každá z těchto sekcí nabízí mnohem více funkcí. Celkový přehled všech podporovaných funkcí lze vidět na obrázku 7.

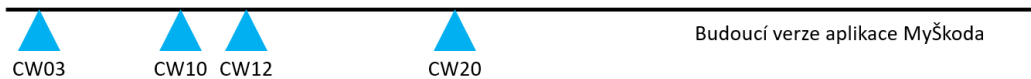
MOD1	MOD2	MOD3	MOD4
Service Scheduling	Vehicle Health Report	Vehicle Health Report	Vehicle Health Report
Online Personalization	Service Scheduling	Service Scheduling	Service Scheduling
Info Call	Online Personalization	Online Personalization	Online Personalization
Parking Position	Info Call	Info Call	Breakdown call
Vehicle Status	Online Ventilation	Predictive maintenance	Info Call
Driving Data	Online Auxiliary Heater	Online Ventilation	Predictive maintenance
Speed Notification	Parking Position	Online Auxiliary Heater	Parking Position
Area Notification	Honk & Flash	Parking Position	Vehicle Status
Car Feedback	Vehicle Status	Honk & Flash	Pay to Park
Doors & lights	Driving Data	Vehicle Status	Car Feedback
Notifications	Speed Notification	Driving Data	Doors & lights
Service Partner add-on	Area Notification	Speed Notification	Notifications
Trip Planner	Car Feedback	Area Notification	Service Partner add-on
Vehicle Details	Doors & lights	Pay to Park	Vehicle Details
Wizards	Notifications	Car Feedback	Departure Planner
Manuals	Remote Trip Statistics	Doors & lights	Wizards
Application start	Service Partner add-on	Notifications	Manuals
Remote Battery Charging	Trip Planner	Remote Trip Statistics	Remote Profiles and Timer (RPT)
Remote Departure Timer	Vehicle Details	Service Partner add-on	Online Anti-Theft Alert
Remote Air Conditioning	Wizards	Trip Planner	Remote Battery Charging
Enrolment	Manuals	Vehicle Details	Remote Departure Timer
	Application start	Wizards	Remote Air Conditioning
	Lock & Unlock	Manuals	Privacy mode
	Online Anti-Theft Alert	Remote Profiles and Timer (RPT)	Enrolment
	Car Access	Application start	Identification, Authentication and Authc
	Remote Departure Timer	Lock & Unlock	
	Remote Air Conditioning	Online Anti-Theft Alert	
	Privacy mode	Car Access	
	Enrolment	Remote Battery Charging	
	Identification, Authentication and Authc	Remote Departure Timer	
		Privacy mode	
		Enrolment	
		Identification, Authentication and Authorization(IAA)	

Obrázek 7: Přehled funkcí MyŠkoda

Zdroj: vlastní snímek obrazovky sešitu Excel

MyŠkoda si prošla mnohými změnami a verzemi v průběhu času. U zákazníků se nyní nachází sedmičková verze MyŠkoda s označením 7.X.X, kde první X je nějaká nová nadstavba sedmičkové verze a druhé X je označení například nějakého hotfixu. Při přechodu na sedmičkovou verzi došlo k implementaci podpory pro třetí generaci vozů. Byly také zpřístupněny funkce jako Pay to Park, Pay to Fuel a funkce digitálního certifikátu, na kterou je zaměřena praktická část této práce v kapitole 4. Ve verzi 7.2.0 došlo k integraci Powerpassu zmíněného v kapitole 3.1.4. Poté ve verzi 7.3.0, což je nejnovější verze MyŠkoda, došlo k implementaci podpory pro první a druhou generaci vozů. Tímto MyŠkoda podporuje všechny vozy a stala se jedinou aplikací pro komunikaci s autem. Aplikace MyŠkoda Essentials zmíněná v kapitole 3.1.4 je tím pádem nepotřebná a bude jí zrušena podpora. Na obrázku 8 je znázorněno postupné zveřejňování verzí MyŠkoda, kde CW znamená kalendářní týden, kdy verze byla zveřejněna. Jak již bylo zmíněno, 7.3.0 je nejnovější verzí, ale Škoda Auto a.s. má naplánované nové verze do budoucna.

#### Timeline sedmičkových verzí v roce 2024



Obrázek 8: Timeline sedmičkových verzí v roce 2024

Zdroj: vlastní

### 3.3 Popis procesu manuálního testování aplikace MyŠkoda

Hlavní náplní zde popisovaného manuálního testování je testování MyŠkoda aplikace. Při testování se používá takzvaná RC verze aplikace, při nálezů chyby se pro testování užívá i debug verze aplikace. RC znamená Release Candidate a je to verze aplikace připravená pro kvalitativní testování viz kapitola 2.1. Z pohledu vývoje by tato verze neměla obsahovat žádné bugy, neboli chyby, a měla by být stabilní.

V tomto ohledu se liší automatizované testování od manuálního, a to v tom, že manuální testování je aplikováno pouze na oficiálně schválené verze aplikace, které jsou však stále testovací a k zákazníkovi se nedostanou. Toto je rozdíl oproti kapitole 2.1, který byl objeven v průběhu praxe. Pro Android se tyto verze aplikace získávají z takzvaného alfa kanálu internetového obchodu Google Play a pro IOS se získávají z aplikace jménem TestFlight, což je oficiální aplikace od společnosti Apple pro testování aplikací, kde jsou pak zpřístupněny schválené verze aplikace. Při nalezení nějaké chyby se poté testuje takzvaná debug verze, viz úvod kapitoly 3.3, kde je přístup k podrobnějším logům z aplikace, a proto je lepší pro získávání zpětných informací, jak se tato chyba vyskytnula a co je její příčinou.

Testování vychází z takzvaných use casů neboli testovacích scénářů. Jsou to postupy, které jsou připravené lidmi z vývoje od začátku vývoje aplikace. Měly by být napsány tak, aby se přiblížily akcím zákazníka co nejpřesněji. Tyto testy, jsou poté upravovány pro vlastní použití. Většinou částí testování jsou poté také takzvané explorativní testy. Při těchto testech se pořád následují z části use casey, ale při testování je určitá volnost a možnost pohybu v aplikaci. Tím se liší testování na kvalitě od vývoje. Testování na kvalitě využívá kvalitativní testování. Vývoj následuje tyto testovací scénáře přesně krok po kroku narozdíl od kvalitativního testování. V tomto se manuální testování liší od automatizovaného testování, kde

automaty postupují přesně krok po kroku podle testovacích scénářů. Manuálním testováním je možno odhalit jiné chyby aplikace, které testovací scénáře nebraly v úvahu.

Při nalezení bugů se musí bug reportovat. Před reportováním se musí stáhnout debug verze a získat logy z aplikace. Tato chyba se pak musí replikovat na debug verzi. Pokud se nepovede tuto chybu replikovat na debug verzi, musí se vytáhnout logy z RC verze. Po získání všech těchto informací ohledně chyby se může začít s reportováním. Reportování funguje pomocí takzvaných ticketů v softwaru JIRA viz kapitola 2.1. Stejným způsobem se reportují i chyby nalezené při automatizovaném testování. Ticket musí obsahovat veškeré informace ohledně chyby. Mimo jiné musí obsahovat verzi aplikace, čeho se to týká nalezená chyba, jaké platformy, verze softwaru auta, popis vlastního postupu, jaký byl výsledek, jaký byl očekávaný výsledek a přílohy, což jsou videa a snímky obrazovky. Poté je reportován JIRA ticket.

Po vytvoření ticketu přichází jeho analýza zodpovědnými lidmi z vývoje. Jsou reportovány 2 typy bugů, čistě bug a external bug. Bug je využíván vývojovým týmem aplikace MyŠkoda a external bug je určen pro všechna ostatní oddělení Škoda Auto a.s. External znamená vnější neboli mimo vývoj. Bugy se rozlišují trochu jinak pro vývojový tým MyŠkoda a pro oddělení kvality. Vývoj má čtyři různé bugy narozdíl od kvality, která má bugy tři. Tyto bugy jsou označovány skoro stejně, ale nesou jinou závažnost chyby. Oba, jak vývoj, tak kvalita, mají „Áčko“, „Béčko“ a „Céčko“, ale vývoj má navíc i „Déčko“. Je také možno tyto bugy jednoduše označit jako A, B, C a D. Z pohledu kvality se tyto bugy rozdělují takto. Pod A spadají veškeré bugy, kde nefunguje jakákoliv funkce aplikace a zákazník nemá jiný přístup k této funkci, nebo také například, když aplikace spadne. Vývoj se v tomto liší. Vývoj má totiž tyto chyby rozděleny do dvou. Bug A se liší v tom, že pro ně A znamená pouze pád aplikace a B jsou, jak již bylo zmíněno v této kapitole 3.3, nefunkční funkce. Pro kvalitu pod kategorií B spadají bugy, kde funkce nemusí fungovat, ale zákazník se může k výsledku dostat i jiným způsobem. Pro vývoj by byl tento bug označen jako C. Jako bug C pro kvalitu se většinou označují jen vizuální bugy, jako je například překlep v textu nebránící člověku v jeho pochopení. Pokud by ale byla

chyba v právních textech, byla by tato chyba zařazena do vyššího stupně bugu. Pro vývoj by bylo použito označení D.

Jednou týdně se potom řeší veškeré bugy na týdenním meetingu jménem Bug Status, kde všechny zodpovědné oddíly probírají různé bugy a jejich prioritu odstranění. Po testování, odreportování a odstranění hlavních bugů přichází go meeting. Na tomto meetingu dochází k odsouhlasení, zda je aplikace připravena ke svému zpřístupnění. Všichni zodpovědní stakeholderi, diskutují o možném uvolnění aplikace pro veřejnost. Pokud všichni stakeholderi odsouhlasí zveřejnění aplikace, tak dochází k její přípravě pro vydání. Po jejím připravení aplikace není hned přístupná všem zákazníkům najednou. Je zpřístupňována takzvaným graduálním releasem. To znamená, že Apple nebo Google vydávají aplikaci postupně určitému procentu zákazníků. Například u společnosti Apple se začíná na jednom procentu, kde má k aplikaci přístup jen jedno procento zákazníků. Toto procento se postupně navyšuje. Apple i Google mají svoje způsoby, jakými určují, jakému zákazníkovi je aplikace zpřístupněna a jakému ne.

## **4 Automatizace testování digitálního produktu aplikace MyŠkoda**

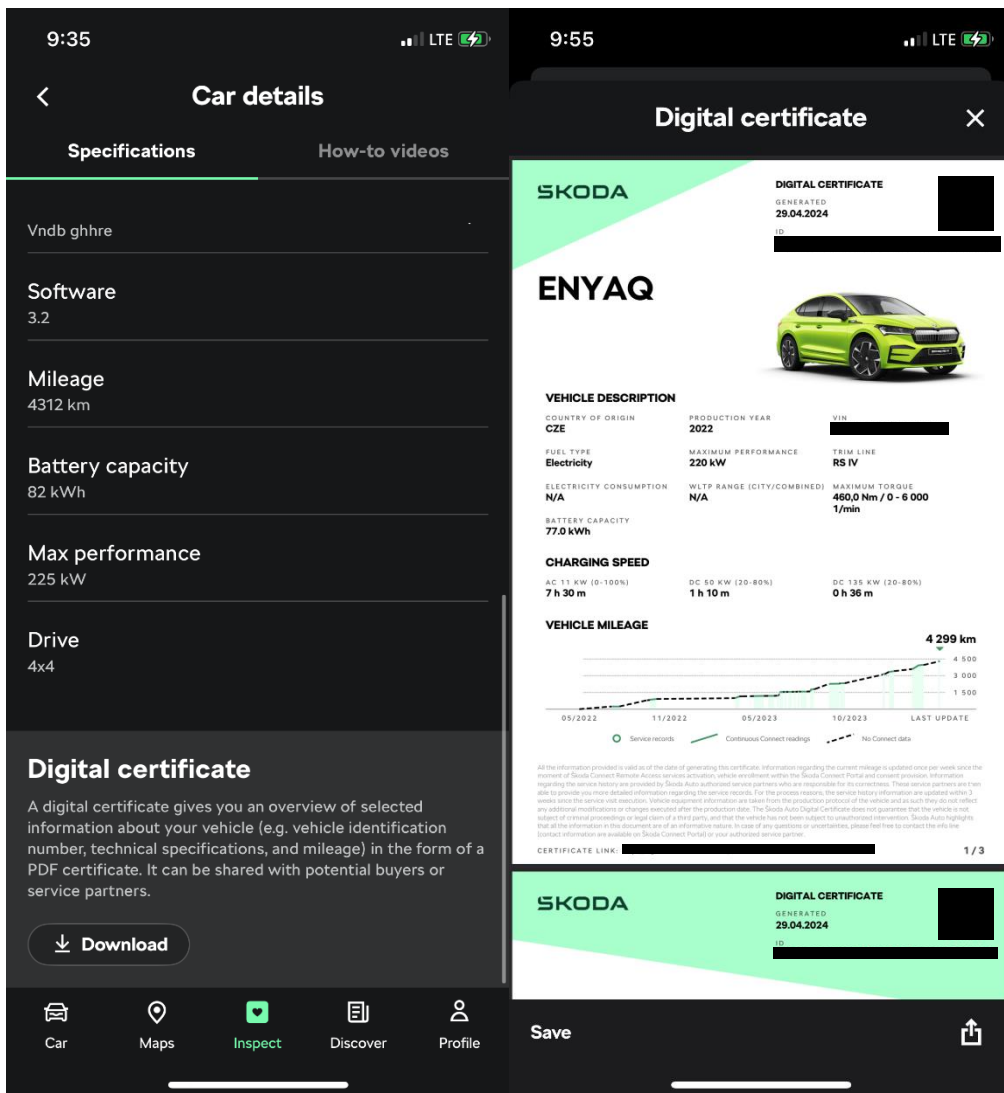
Automatizované testování frontendu digitálních produktů a aplikací se provádí za použití automatizovaných testovacích skriptů a nástrojů k provádění předdefinovaných testovacích scénářů a ověřování chování mobilní aplikace MyŠkoda. Tato kapitola má za cíl představit, jak funguje automatizované testování v reálném prostředí. Tato kapitola nebude obsahovat všechny kroky popsané v kapitole 2.1, protože během praxe ve Škoda auto a.s. se zjistilo, že praxe se odlišuje od teorie.

### **4.1 Analýza pro vývoj a popis nového testovacího scénáře**

Při vydání nového RC (Release Candidate), viz kapitola 3.3, kde přichází nové funkce do aplikace MyŠkoda oproti její původní verzi, je zapotřebí zhodnotit, zda nové funkce musejí být testovány, a pokud ano, zda jsou testovatelné. O tomto rozhoduje oddělení kvality. Po vyhodnocení tohoto prvního kroku se zadá úkol analytikům pro vytvoření testovacího scénáře. Tyto testovací scénáře jsou ukládány do podnikových databází Jira a Confluence. Veškeré testování a vytváření testovacích scénářů probíhá jedna ku jedné jak na platformě iOS, tak na platformě Android. Zaměřením této kapitoly bude sekce aplikace MyŠkoda Prohlídka (Inspect), viz kapitola 3.2, protože se zde nachází nová funkce Digitální certifikát (Digital certificate).

Digitální certifikát je součástí záložky Podrobnosti o vozidle (Car details) v sekci Prohlídka. Smyslem této funkce je možnost zákazníka vygenerovat certifikát ve formě PDF. Tento Certifikát dává přehled o vybraných informacích vozidla, např. identifikační číslo vozidla, technické specifikace nebo ujeté kilometry. Toto umožňuje lehčí sdílení informací o vozidle s potenciálním kupcem nebo servisními partnery. Z důvodu, že je tato funkce důležitou součástí aplikace, je proto vhodným kandidátem na automatizované testování a tím pádem i na analýzu pro vytvoření testovacího scénáře.

Když byla funkce identifikována jako důležitá, dalším krokem je zjistit, zda bude automatizovatelná. Na obrázku 9 je vidět, že se jedná pouze o text a tlačítko Stáhnout (Download). Kontrola textu a klikání na tlačítko je jednoduše automatizovatelné. Při kontrole textu dochází k porovnání textu v aplikaci a textu v testovacím scénáři proto se musí jednat o stálý text, který se nemění. Testovací scénáře jsou vytvářeny na konstantní text, který se nemění a zůstává stejný, protože test není autonomní a nedokáže se přizpůsobovat dynamickému textu a musí být manuálně upravován. Kdyby se jednalo o text, který se často mění, jednalo by se o neautomatizovatelný test, protože by nemělo smysl automatizovat něco, co se musí pořád upravovat. Za neautomatizovatelné by se také považovalo například kontrolování obrázků a tím není myšlen výskyt, ale obsah obrázku. Na toto testování není ještě technologie připravena.




Obrázek 9: Sekce car details a vygenerovaný certifikát

Zdroj: vlastní snímek obrazovky aplikace MyŠkoda

Při vytváření testovacího scénáře musí analytik vymyslet postupné kroky neboli takzvané *stepy*, podle kterých bude poté vývojář vyvíjet daný test. Jak je vidět na obrázku 10, tak každý krok musí být řádně označený. Pro označení se používá sedm různých sloupečků. Za prvé musí být každý krok očíslen vzestupně od jedničky až po počet potřebných kroků. Na obrázku 10 je to vidět ve sloupci jako číslo kroku (Step no) a protože se jedná o první krok v testu tak je tento krok označen jedničkou. Za druhé musí být tento krok nějak pojmenován. Pro to slouží sloupec s názvem krok testu (Test step). Tento sloupec znázorňuje, o co v daném kroku jde. V tomto případě jde o přihlášení do MyŠkoda aplikace (Login to MSP). Za třetí musí v tomto kroku být jednoduše popsáno co má tento krok udělat. Tady jde kliknutí na tlačítko pro přihlášení (Click on Login button). Ve čtvrtém sloupci jde o přidání



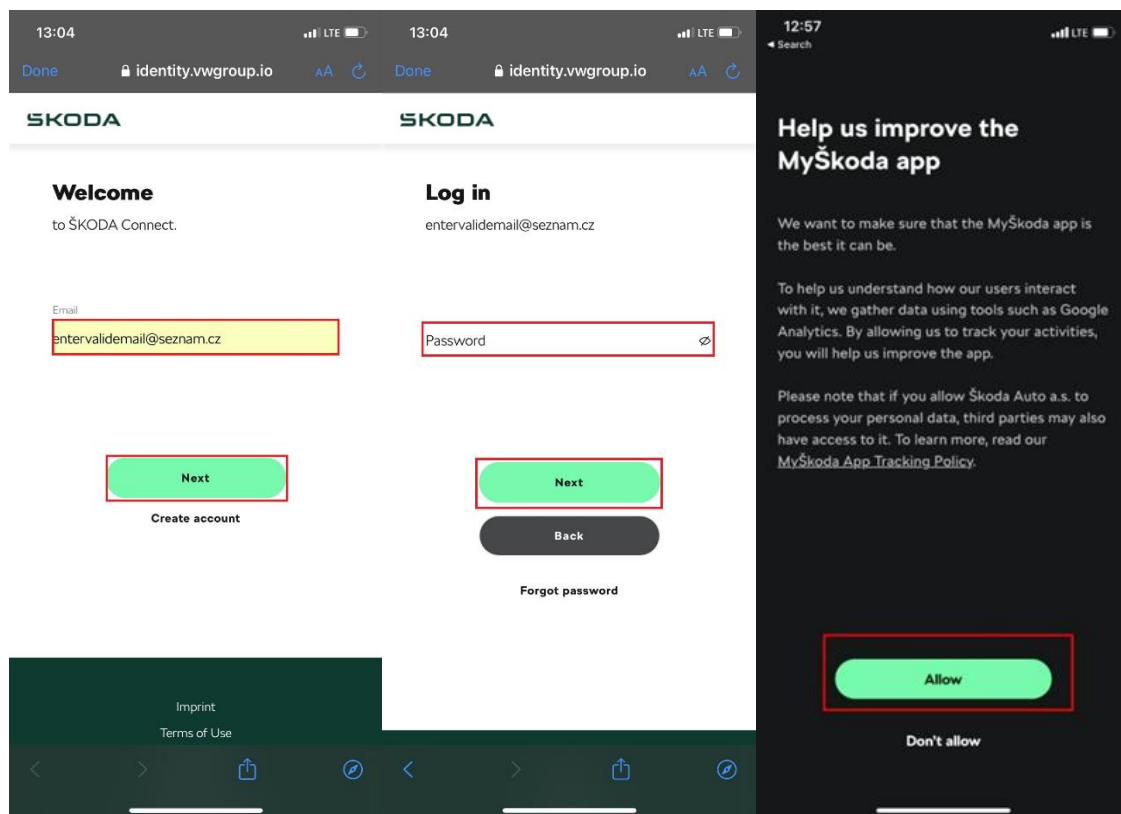
snímku obrazovky nebo takzvaného screenshotu. Jde o vizuální znázornění, pokud například lidé, kteří nejsou seznámeni s touto tematikou, tak aby bylo na první pohled zřejmé, o co v tom kroku jde. Každý snímek obrazovky má vyznačenou část červeným obdélníkem. To je další vrstva, která napomáhá k přehlednosti kroku a ukazuje, kam přesně bude test klikat nebo jaký text bude kontrolovat. Za páté se jedná o sloupec s poznámkou (Note). Toto je první a poslední sloupec, který není povinný. V tomto sloupci se většinou uvádí doplňující informace, které by se jinak nikam jinde nevěšly. Co se týče šestého sloupce ten je spojený se sedmým sloupcem. Každý krok musí být zařazen do jedné tří kategorií. Jsou to kategorie času odezvy. Tyto kategorie jsou rozděleny podle písmen A, B a C. Kategorie A je nejkratší z těchto kategorií, a to jen třicet sekund. Kategorie B je sto dvacet sekund a kategorie C je tři sta sekund. Každý krok má určitou náročnost časové odezvy a je zařazen do jedné z těchto kategorií. Jak je vidět na obrázku 10, tak byl tento krok zařazen do kategorie A s třiceti sekundovou dobou čekání, protože klikání na tlačítko přihlášení by nemělo být časově náročné. Kdyby náhodou došlo k překročení tohoto limitu tak by poté test spadl, ale o tom více v následující kapitole. Šestý sloupec je proto pojmenován čas čekání na odezvu, kde je vyplněn čas v sekundách a sedmý sloupec je pojmenován kategorie hodnocení, kde je vyplněno jedno z písmen A, B nebo C již zmíněno v kapitole 4.1. Tímto způsobem se připravují a označují všechny kroky testu.

Step no	Test step	Procedure	Attachments	Note	Waiting time for a response	Evaluation category
1	Login to MSP	Click on <b>Login</b> button.			30	A

*Obrázek 10: Ukázka prvního kroku testovacího scénáře*

Zdroj: vlastní snímek obrazovky aplikace MyŠkoda

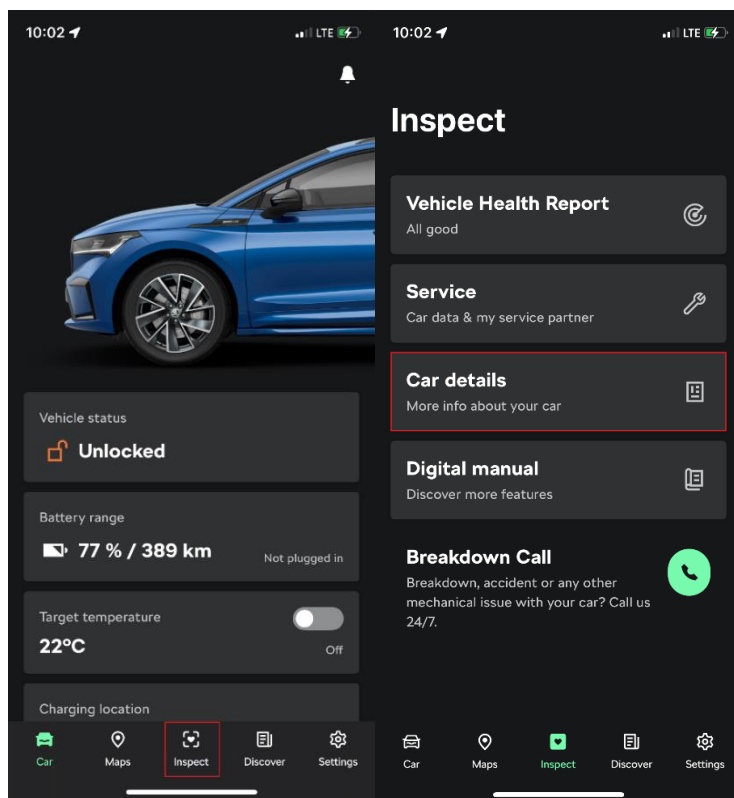
Jak již může být zřejmé, všechny testy začínají přihlášením neboli loginem. Instalace a spuštění aplikace nejsou součástí analýzy, protože se o to stará platforma Browserstack. Browserstack je platforma na které se spouštějí vzdáleně testy na reálných zařízeních, kde poté automaticky probíhá reinstalace aplikace a její následné spuštění. O této tématice bude zmíněné více v další podkapitole. Pro otestování požadované funkce se musí test proklikat aplikací na místo kde se daná funkce nachází. Jak již bylo zmíněno, tak test začíná loginem. U každého testu je login stejný. Může se pouze lišit podle toho, na jakém autě se testuje a tím pádem se bude muset přihlásit na jiný účet. Po kliknutí na tlačítko *log in* musí test zadat email a heslo. Ukázku je možno vidět na obrázku 11. Po přihlášení, než se se test dostane na domovskou sekci, kterou je sekce *vůz*, tak musí test ještě odsouhlasit zlepšování MyŠkoda app, což lze vidět také na obrázku 11.



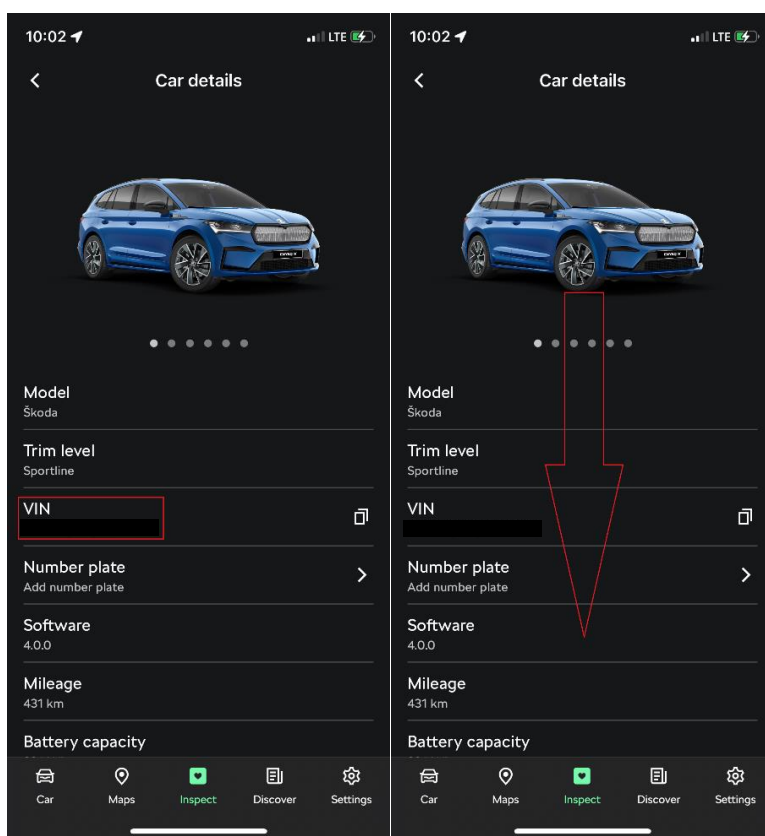
Obrázek 11: Přihlašování do MyŠkoda aplikace

Zdroj: vlastní snímek obrazovky aplikace MyŠkoda

Ze sekce *zlepšování MyŠkoda app* se test musí proklikat do sekce *prohlídka*, kde se nachází záložka *podrobnosti o vozidle*, tam se nachází funkce *digitální certifikát*, která má být testována. Poté v záložce *podrobnosti o vozidle* test zkopíruje VIN kód, protože při přípravě testovacího scénáře bylo odhaleno, že bude potřeba zkontrolovat, zda se digitální certifikát vygeneroval se správným VIN kódem a k tomu slouží tento krok. Dalším krokem je zascrollovat na spodek stránky, kde se nachází funkce na testování. Všechny tyto kroky mohou být vidět na obrázcích 12 a 13.

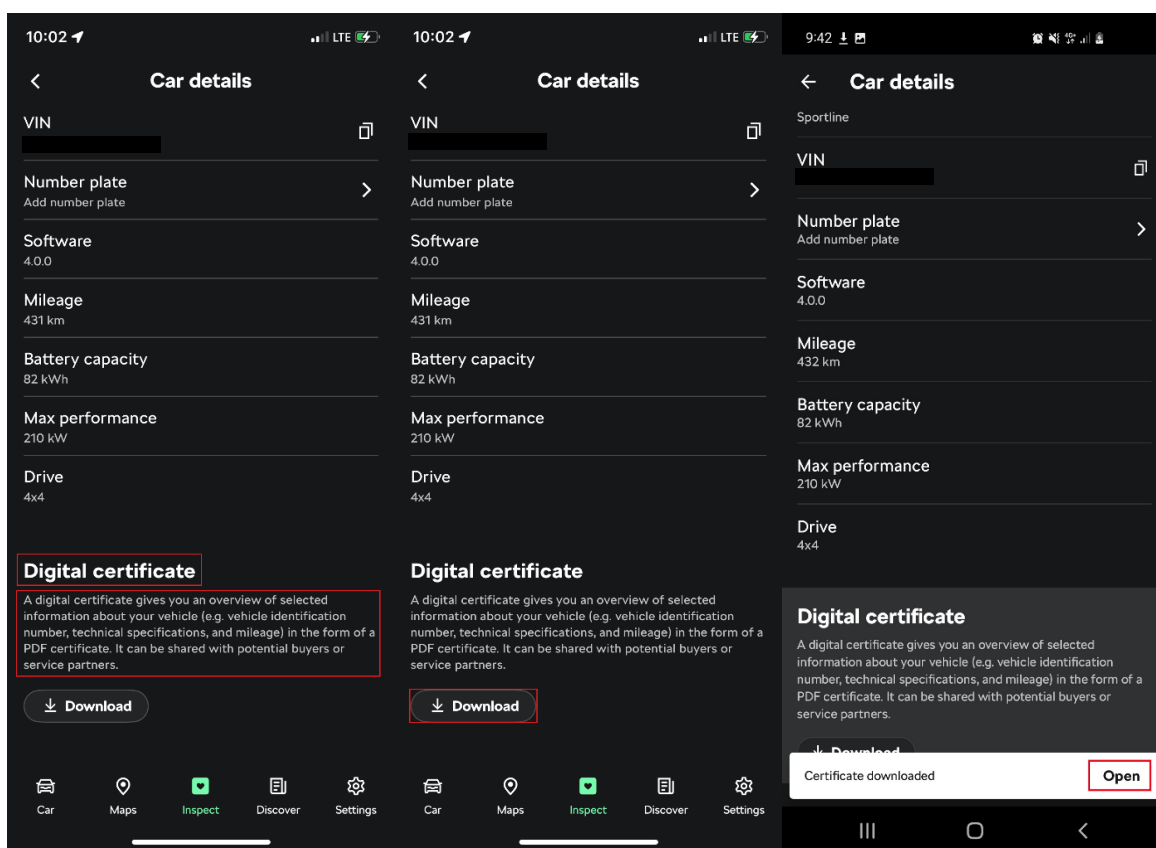


Obrázek 12: Tlačítko inspect a car details  
Zdroj: vlastní snímek obrazovky aplikace MyŠkoda



Obrázek 13: Kopírování VIN kódu a swipe  
Zdroj: vlastní snímek obrazovky aplikace MyŠkoda

Po tom, co se test dostane na spodek obrazovky, může začít kontrolovat digitální certifikát. Počátečním krokem je zkontrolování textu. Pokud nenastane chyba, test pokračuje kontrolou a kliknutím na tlačítko stáhnout pro stažení a otevření digitálního certifikátu. Test se v dalším kroku liší na Androidu oproti iOS. Oba testy čekají na stáhnutí digitálního certifikátu, ale na iOS dochází k jeho automatickému otevření na rozdíl od Androidu, kde test musí klikat na tlačítko otevřít (Open), které vyskočí s pop upem (vyskakující lištička), kde je oznámeno uživateli, že byl certifikát stažen. Tyto kroky je možno vidět na obrázku 14, kde na pravé straně je krok navíc pro Android oproti iOS.

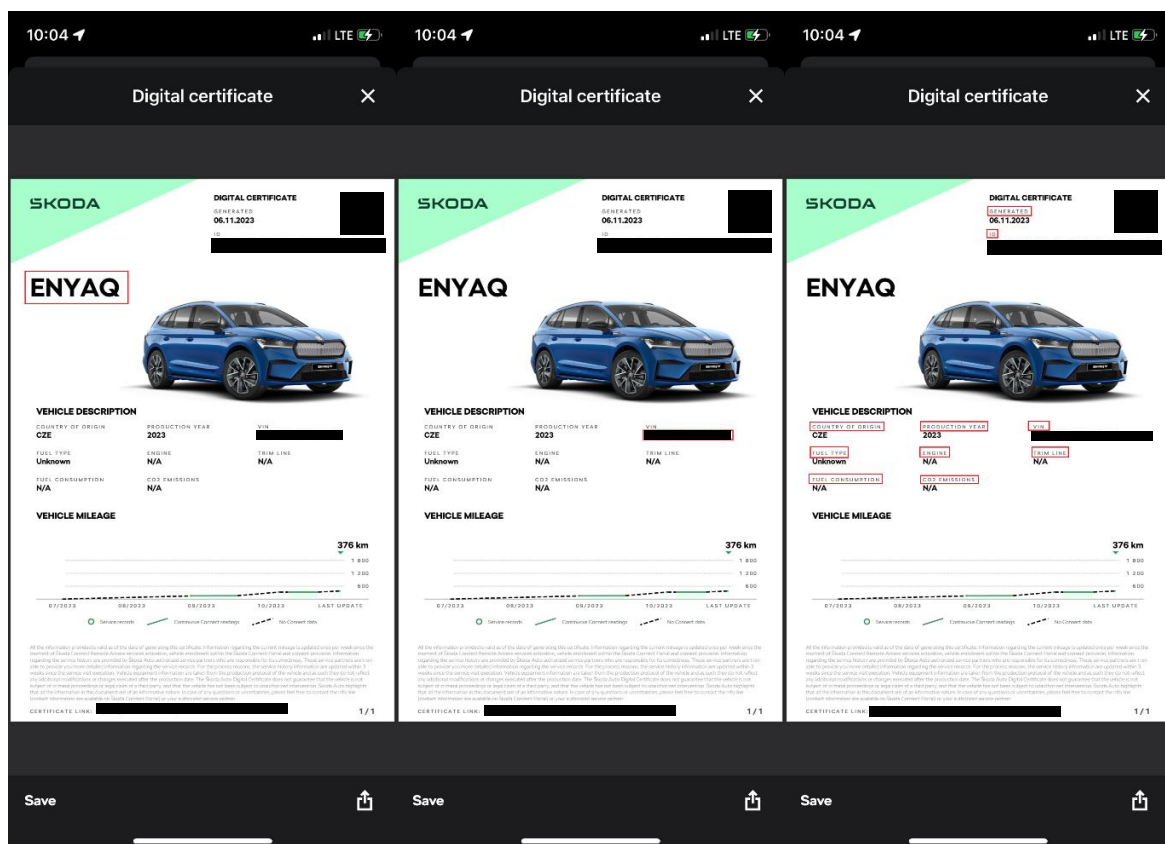


Obrázek 14: Text a tlačítko pro stažení a otevření digitálního certifikátu

Zdroj: vlastní snímek obrazovky aplikace MyŠkoda

Následně přichází poslední část testu, a to je kontrola vygenerovaného digitálního certifikátu. V prvním z těchto kroku kontroluje test, zda se vygenerovalo jméno auta správně. Je to to ten hlavní největší text. Následuje kontrola již zmiňovaného VIN kódu, kde test porovnává VIN kód z podrobností o vozidle s VIN kódem z vygenerovaného certifikátu. Posledním krokem tohoto testu je kontrola stálých textů například textu vygenerováno (generated), pod kterým je uvedeno, kdy byl

certifikát vygenerován nebo země původu, typ paliva a tak dále. Všechno je znázorněno na obrázku 15.



Obrázek 15: Vygenerovaný digitální certifikát  
Zdroj: vlastní snímek obrazovky aplikace MyŠkoda

## 4.2 Vývoj předpřipraveného testu

Vývoj testu začíná získáním elementů v aplikaci. Tyto elementy se získávají pomocí aplikace, která se nazývá Appium Inspector. Po připojení telefonu k počítači nebo připojení na vzdálený telefon Appium Inspector projíždí aplikaci a nachází veškeré elementy, které se v aplikaci nachází. Tyto elementy je potřeba najít z důvodu určení co test hledá a kam má klikat. Tyto jsou označeny unikátním kódem, který se nazývá Xpath. Jednoduše řečeno se jedná o adresu elementu. Tyto Xpathy se liší jak pro Android, tak pro iOS. Vývojář si tedy musí projet aplikaci a vytáhnout si veškeré elementy, které bude potřebovat pro programování testu. Tyto elementy se poté ukládají do seznamu elementů v Pythonu. Odkud se pak na ně odkazuje v kódu. Na obrázku 16 je možno vidět ukázkou kódu seznamu elementů.

```
# INSPECT
inspect = (AppiumBy.XPATH, '//android.widget.TextView[@text="Inspect"']
driving_data = (AppiumBy.XPATH, '//android.widget.TextView[@text="Driving data"']
icon_vhr = (AppiumBy.ACCESSIBILITY_ID, 'Vehicle Health Report icon')
vehicle_health_report_btn = (AppiumBy.XPATH, '//android.widget.TextView[@text="Vehicle Health Report"']
service_btn = (AppiumBy.XPATH, '//android.widget.TextView[@text="Service"']
car_details_btn = (AppiumBy.XPATH, '//android.widget.TextView[@text="Car details"']
```

Obrázek 16: Kód seznamu elementů

Zdroj: vlastní snímek obrazovky aplikace MyŠkoda

Po získání veškerých elementů a jejich zapsání do seznamu může začít programování testu. Na začátku kódu musí vždy být načtení potřebných funkcí z modulů. Tyto funkce byly již vytvořeny mými předchůdci. Každý test potřebuje funkci `sq_appium_test` a `Stage`. První funkce zaručuje správné spouštění testu a druhá funkce se používá k rozdělení testu do různých stepů jako v analýze. Poté se tu načítají funkce z modulu `helpers`. Modul `helpers` obsahuje funkce pro získávání elementů a také funkce pro pohyb v aplikaci, protože se v aplikaci nedá jen klikat, ale musí občas také posouvat obrazovka a k tomu slouží tyto funkce. Z `helpers` se importuje funkce `login`, která slouží k přihlášení do aplikace. Jelikož každý test obsahuje přihlášení do aplikace, tak byl celý `login` dán do jedné funkce, kterou lze načíst. Poté se načítá funkce `getImportElements`, která načítá již zmíněný seznam elementů. Dále `getPlatform` získává informace o tom, zda se jedná o platformu `Android` nebo `iOS`. Pak tu jsou dvě funkce pro nalézání a získávání hodnoty elementů. První funkcí je `getElement`, jak již jméno napovídá, jedná se o funkci nalezení elementu a poté to je `getAttribute`, ta se využívá pro získání hodnoty elementu po jeho nalezení. V neposlední řadě se načítá funkce `swipeToElement`. Ta je využívána pro swipování neboli posouvání pro obrazovce k určitému elementu. Importování funkcí je možné vidět na obrázku 17.

```
from sq import sq_appium_test
from stage import Stage
from helpers import (
    login,
    getElement,
    getImportElements,
    getPlatform,
    getAttribute,
    swipeToElement)
```

Obrázek 17: Importování funkcí

Zdroj: vlastní snímek obrazovky aplikace MyŠkoda

Po načtení veškerých potřebných funkcí začíná psaní hlavní části kódu. Prvním krokem je uložení funkcí `getImportElements` a `getPlatform` do proměnných z důvodu přehlednosti kódu. Po uložení těchto dvou funkcí do proměnných přichází první stage. Tento stage obsahuje část s `loginem`, protože `login` je prvním krokem v testu. Stage by se dal označit také jako krok v naší analýze. Každý stage musí obsahovat jméno kroku, proměnnou `driver` do, které se nahrává jméno daného kroku, které se například vypisuje po chybě testu pro přehled, kde se test zasekl a písmeno pro označení jak dlouho daný krok má trvat. Stejně jako v analýze. V této stage bude pouze funkce `login`, která obsahuje vše potřebné k přihlášení do aplikace. Po přihlášení následuje kliknutí na tlačítko `inspect`. V nové stage půjde o kliknutí na tlačítko `inspect`. Toho jde dosáhnout za pomoci funkce `getElement` a `click`. Element musíme nahrát do určité proměnné, kterou si nazveme `inspect`. Do `inspectu` se poté element nahraje pomocí `getElement` a to tak, že `getElement(driver, elements.insepct_btn)`. Proměnná `driver` byla již zmíněna a `elements.inspect_btn` je funkce pro importování adresy elementu v tomto případě pro tlačítko `inspect` v našem seznamu elementů. Poté co test tento element najde, musí na něj kliknout. O to se stará funkce `click`. Kde vezmeme naší proměnnou `inspect` do které jsme nahráli adresu `inspect` tlačítka a pomocí funkce `click` na ní klikneme. Kód vypadá takto `inspect.click()`. Po tom co test klikne, se musí ještě zkontrolovat, zda se test dostal na správnou stránku. Kontroluje se to pomocí `getElement`, kde se do proměnné nahrává, zda se na stránce nachází text `inspect`, který je nadpisem této



stránky. A tímto způsobem se postupuje v celém testu. Dalším stagem je kliknutí na tlačítko car details. Takže si zase najedeme, kde se tento element nachází, pomocí getElement, klikneme na něj a poté zkontrolujeme, zda se test dostal na správnou stránku. Dalším stagem je kopírování VIN kódu pro pozdější porovnání. Stejně jako u předchozích stagů se najde pozice VIN kódu pomocí getElement. Poté se se kód najde, získává se jeho hodnota. Tato hodnota se získá pomocí funkce getAttribute. Kód vypadá takto. `vin_text = getAttribute(vin, platform)`. Hodnota se nahraje do proměnné `vin_text`. Poté se ve funkci vyplňuje, z čeho se získává hodnota. V tomto případě se jedná o naši proměnnou s VIN kódem a proměnnou `platform`, protože ukládání hodnoty se liší, jak pro Android a iOS. V dalším stage jde o kontrolu textu u digitálního certifikátu. V tomto případě, kdy se nachází text úplně na spodku obrazovky, musí test swipnout. K tomu využíváme funkci `swipeToElement`. Zde se zadá jen element, ke kterému se má swipnout. Tímto elementem je tlačítko pro stažení. Kód vypadá takto `swipeToElement(driver, elements.download_btn_cert)`. Poté co test posune obrazovku, kontroluje se text za pomoci getElement. Následuje stage ve kterém se stahuje a otevírá digitální certifikát. Stejně jako v předchozích stagích se najde element pomocí getElement a klikne se na něj. Přichází zde, ale kód navíc. Z důvodu odlišností Android a iOS se zde musí dát podmínka. Pokud se jedná o Android, musí se ještě hledat tlačítko *open* pro otevření certifikátu. Toto u iOS neplatí, protože se certifikát otevírá automaticky. Takže pokud se `platform == "android"`, tak se hledá tlačítko *open* pomocí getElement a kliká se na něj. Po otevření digitálního certifikátu následuje poslední stage. Kontrolují se zde hodnoty a jejich správné vygenerování. Prvním krokem je získání hodnoty textu. Protože hodnoty digitálního certifikátu jsou v jednom dlouhém textu a tyto hodnoty se poté musí v tomto textu hledat. Stejně jako VIN kódu se získá celý text pomocí getElement. Po získání textu se kontrolu za pomoci funkce assert. Assert se používá k ověření, že určitá podmínka je pravdivá. Kód vypadá takto `assert vin_text in pdf_text, "ERROR -VIN is wrong"`. Proměnná `vin_text` je naše hodnota VIN kódu, `pdf_text` je hodnota celého textu našeho certifikátu. Takže pokud by se `vin_text` nenacházel v `pdf_text` tak by se vypasala zpráva "ERROR – VIN is wrong". Tímto končí kód testu. Ukázka kódu je možno vidět na obrázku 18.

```
with Stage('Go to Inspect', driver, 'A'):
    inspect = getElement(driver, elements.inspect_btn)
    inspect.click()
    inspect_page = getElement(driver, elements.inspect)
```

Obrázek 18: Ukázka hlavního kódu  
Zdroj: vlastní snímek obrazovky aplikace MyŠkoda

## **5 Přínosy automatizace ve vývoji digitálních produktů pro podnikové procesy a ekonomiku**

Pro oddělení kvality je důležité testovat takovým způsobem, kde se přiblíží co nejvíce k pohledu zákazníka. Tímto způsobem získají co nejpřesnější data a lepší kvalitu produktu. Tato kapitola se bude zabírat přínosy automatizovaného testování viděných na stáži ve Škoda Auto a.s. a jejich dopad na procesy a ekonomiku podniku.

### **5.1 Přínosy automatizovaného testování pro kvalitnější produkt a procesy firmy**

Jak již bylo naznačeno v úvodu kapitoly 5.0, oddělení kvality se snaží přiblížit k pohledu zákazníka při testování. Pomocí automatizovaného testování je poté možno získat více dat ohledně aplikace a chování zákazníka. Tyto data mohou být poté použity pro zlepšení a upravení aplikace a k jejímu zkvalitnění. Mezi tyto data patří například nálezy chyb. Konzistentní a opakované spouštění testů má za výsledek nárůst nálezu chyb, a protože automatizované testy jsou rychlejší, než manuální testy dochází také k rychlejšímu nálezu těchto chyb. To má za následek rychlejší opravu chyb a tím pádem i produkt, který obsahuje menší procento chyb. Dále během stáže bylo vidět, že testy napomáhají kontrole funkčnosti aplikace. Funkce, které fungují správně a bezchybně vedou ke spokojenějším zákazníkům a vyšší loajalitě. Z pohledu kvality automatizované testy také zachovávají větší objektivnost vůči problému. Různí testeři budou mít různé názory a mohou vyhodnotit chybu odlišně, ale automatizované testy vyhodnotí problém vždy stejně. Tím se eliminuje lidská chybovost. Dalším přínosem, který byl vyzpozorován je, že testy mohou běžet bez přestávky a pokrýt delší časový úsek. Tímto mohou zajistit stabilitu aplikace v delším časovém horizontu. Všechna tyto data jsou cenná a napomáhají k identifikaci oblastí pro zlepšení toto vše může vést ke zvýšení kvality produktu a optimalizaci vývojových procesů.

## 5.2 Dopad automatizovaného testování na ekonomiku

Na první pohled ekonomické přínosy automatizovaného testování firmu Škoda Auto a.s. nejsou úplně zřejmé a jejich vyčíslení je skoro nemožné. Je potřeba se zamyslet co vlastně automatizované testování přináší z pohledu ekonomiky. Při využívání automatizovaného testování nelze jednoduše říct, že při nasazení automatizovaného testování na X procent bude zisk tolik a tolik procent. Takové to vyčíslení je velmi těžké, ale existují způsoby, kterým automatizované testování přispívá ekonomice podniku. Jedním přínosem je zefektivnění využití času. Efektivnější využití času se rovná většímu objemu udělané práce tím pádem i k většímu zisku. Když automatizované testování udělá určitou práci za testera může se tester poté soustředit a alokovat svůj čas na jiné potřeby. Zároveň, jak bylo zmíněno v kapitole 5.1, automatizované testy mohou běžet i přes noc. Takto automatizované testy dále zefektivňují časové využití a možnost zvýšení zisku. Tohle jeden ze způsobů, který byl vyzkoušen během stáže, ale hlavním dopadem na ekonomiku produktu je kvalita produktu. V kapitole 5.1 byly popsány způsoby, kterými automatizované testování přispívá k dosažení kvalitnějšího produktu. Čím kvalitnější produkt, tím větší zájem zákazníků koupit si daný produkt. Což znamená větší prodej a ten vzápětí znamená větší zisk. Kvalita produktu má velký dopad na ekonomiku produktu a dalo by se říct, že možná největší. Tyto dva dopady na ekonomiku byly vyzkoušeny během stáže ve Škoda Auto a.s., je ale možné, že existuje více dopadů na ekonomiku.

## Závěr

Digitální produkty se stávají neodmyslitelnou částí života. S rostoucím zájmem o digitální produkty roste i zájem o jejich zpracování a kvalitu. Tato práce představila automatizované testování softwaru jako jedno z řešení pro získání kvalitnějšího digitálního produktu.

V kapitole 1 byl představen základní pohled na digitální produkty, jejich historii a dopad na automotive průmysl. Kapitola 2.1 ukázala, jaké výhody a nevýhody má automatizované testování oproti klasickému manuálnímu testování softwaru a kroky pro správné implementování automatizace. V kapitole 3 došlo k představení produktů Škody Auto a.s. a k popisu procesu manuálního testování ve firmě. Kapitoly 4.1 a 4.2 se zaměřily na ukázkou z praxe ve vývoji testu. V 5 kapitole poté došlo k popsání vyzorovaných přínosů automatizovaného testování a jejich dopad na kvalitu produktu na stáži ve firmě Škoda Auto a.s.

Stálo by také za zamyšlení využití umělé inteligence a strojového učení pro budoucí evoluci automatizovaného testování a plné zautomatizování celého procesu. Automatické vytváření a úprava testů, automatické vyhodnocování a odstranění lidského faktoru by mohlo zefektivnit automatizované testování na maximum. Bohužel umělá inteligence pořád není na takové úrovni, aby to dokázala bez lidské spolupráce, ale mohlo by se jednat o pohled do budoucnosti.

## Seznam použité literatury

- AGU, M.; B. OGBUOKIRI a B. OKWUME, 2016. Comparison of python and java for use in instruction in first course in computer programming. *Transylvanian Review*, vol. 24, no. 7.
- AIRAJ, M., 2017. Enable cloud DevOps approach for industry and higher education. *Concurrency and Computation: Practice and Experience*, vol. 29, no. 9. Dostupné z: <https://doi.org/10.1002/cpe.3937>.
- ALHAMAD, M.; D. Tharam; E. CHANG, 2011. Service Level Agreement for Distributed Services. online. In: *2011 IEEE Ninth International Conference on Dependable, Autonomic and Secure Computing*. Dostupné z: <https://doi.org/10.1109/DASC.2011.172>.
- ARMENGAUD, E.; C. SAMS; G. FALCK; G. LIST; C. KREINER; et al., 2017. Industry 4.0 as Digitalization over the Entire Product Lifecycle: Opportunities in the Automotive Domain. online. In: *24<sup>th</sup> European Conference Systems, Software and Services Process Improvement*. pp. 334–351. Springer. Dostupné z: [https://doi.org/10.1007/978-3-319-64218-5\\_28](https://doi.org/10.1007/978-3-319-64218-5_28).
- AXELROD, A., 2018. *Complete Guide to Test Automation: Techniques, Practices, and Patterns for Building and Maintaining Effective Software Projects*. Berkeley, CA, USA: Apress. ISBN 978-1-4842-3831-8.
- BAO, C., 2022. Inferring the Future of the Electric Vehicle Industry from Tesla's New Business Model. online. In: *Proceedings of the 2022 2nd International Conference on Enterprise Management and Economic Development (ICEMED 2022)*. Dostupné z: <https://doi.org/10.2991/aebmr.k.220603.008>.
- BERAWI, M.; N. SUWARTHA; M. ASVIAL; R. HARWAHYU; M. SURYANEGARA; et al., 2020. Digital Innovation: Creating Competitive Advantages. *International Journal of Technology*, vol. 11, no. 6. Dostupné z: <https://doi.org/10.14716/ijtech.v11i6.4581>.
- BOGDANCHIKOV, A.; M. ZHAPAROV a R. SULIYEV, 2013. Python to learn programming. *Journal of Physics: Conference Series*, vol. 423. Dostupné z: <https://doi.org/10.1088/1742-6596/423/1/012027>.

- CERUZZI, P. E., 2012. *Computing: A Concise History*. Cambridge, Massachusetts, USA: The MIT Press. Dostupné z: <https://doi.org/10.7551/mitpress/9426.001.0001>.
- CHANG, M., 2010. An Agile approach to library IT innovations. *Library Hi Tech*, vol. 28, no. 4, s. 672-689. Dostupné z: <https://doi.org/10.1108/07378831011096303>.
- DERYABINA, G. a N. TRUBNIKOVA, 2021. The Impact of Digital Transformation in Automotive Industry on Changing Industry Business Model. online. In: *IV International Scientific and Practical Conference*. s. 1-7. Dostupné z: <https://doi.org/10.1145/3487757.3490886>.
- DURELLI, V., 2013. *Toward harnessing a Java high-level language virtual machine for supporting software testing*. Ph.D. Thesis. Brazil, São Carlos: Instituto de Ciências Matemáticas e de Computação. Dostupné z: <https://doi.org/10.11606/T.55.2013.TDE-06012014-150025>.
- FELSER, K. a M. WYNN, 2023. Managing the Knowledge Deficit in the German Automotive Industry. online. *Knowledge*, vol. 3, no. 2, s. 180-195. Dostupné z: <https://doi.org/10.3390/knowledge3020013>.
- GARCIA, G., 2008. A New Approach for Satellite Operations and Testing Automation using Python. online. In: *AIAA SPACE 2008 Conference & Exposition*. Dostupné z: <https://doi.org/10.2514/6.2008-7864>.
- GOEL, A.; D. DESHMUKH; M. KUMAR a M. SOI, 2023. Software Quality Assurance. *International Journal for Research in Applied Science and Engineering Technology*, vol. 11, no. 11, s. 1246-1352. Dostupné z: <https://doi.org/10.22214/ijraset.2023.56760>.
- HAIGH, T. (ed.), 2019. *Exploring the Early Digital*. Série: History of Computing. Cham, Switzerland: Springer. Dostupné z: <https://doi.org/10.1007/978-3-030-02152-8>.
- HOLMQUIST, L. E., 2012. Understanding Digital Products. online. *Grounded Innovation*, s. 3-16. Dostupné z: <https://doi.org/10.1016/B978-0-12-385946-4.00001-7>.

- LLOPIS-ALBERT, C.; F. RUBIO a F. VALERO, 2020. Impact of digital transformation on the automotive industry. online. *Technological Forecasting and Social Change*, vol. 162. Dostupné z: <https://doi.org/10.1016/j.techfore.2020.120343>.
- LO, C.; Y. LIN a C. WU, 2015. Which Programming Language Should Students Learn First? A Comparison of Java and Python. online. In: *2015 International Conference on Learning and Teaching in Computing and Engineering*, s. 225-226. Dostupné z: <https://doi.org/10.1109/LaTiCE.2015.15>.
- MONYE, S.; S. AFOLALU; S. LAWAL; O. OLUWATOYIN; A. ADEYEMI; et al., 2023. Impact of Industry (4.0) in Automobile Industry. online. In: *15<sup>th</sup> International Conference on Materials Processing and Characterization*. Dostupné z: <https://doi.org/10.1051/e3sconf/202343001222>.
- PERNUL, G., 2003. Trust for Digital Products. online. In: *E-Commerce and Web Technologies*, s. 1-5. Dostupné z: [https://doi.org/10.1007/978-3-540-45229-4\\_1](https://doi.org/10.1007/978-3-540-45229-4_1).
- PRODUCTFOLIO, 2024. *What is a Digital Product?* online. Dostupné z: <https://productfolio.com/what-is-a-digital-product/>
- PUCHKOVA, E.; E. SOROKOUMOVA; E. CHARDYMOVA a L. TEMNOVA, 2021. Teachers 'and students' ideas about the existing advantages and possible risks of using digital products in the educational environment. *Perspectives of Science and Education*, vol. 53, no. 5, s. 95-109. Dostupné z: <https://doi.org/10.32744/pse.2021.5.7>.
- PUTRANTO, B.; R. SAPTOTO; O. JAKARIA a W. ANDRIYANI, 2020. A Comparative Study of Java and Kotlin for Android Mobile Application Development. online. In: *2020 3rd International Seminar on Research of Information Technology and Intelligent Systems (ISRITI)*, s. 383-388. Dostupné z: <https://doi.org/10.1109/isriti51436.2020.9315483>.
- ROUDENSKÝ, P., 2018. *Kvalita softwaru: Teorie a praxe. 2. upravené a rozšířené vydání*. Prostějov: Computer Media. ISBN 978-80-7402-322-4.
- ŠKODA AUTO a.s., 2023. *View availability of Škoda Connect services*. online. Dostupné z: <https://availability.skoda-auto.com/004/en-com#/>.



- TIAN, J.; A. CHIN a M. KARG, 2016. Digital Services in the Automotive Industry. online. *IT Professional*, vol. 18, no. 5, s. 4–6. Dostupné z: <https://doi.org/10.1109/MITP.2016.93>.
- VAN ROSSUM, G., 2015. The Early Years of Python. online. *Computer*, vol. 48, no. 2, s. 7–9. e-ISSN 1558-0814, ISSN 0018-9162. Dostupné z: <https://doi.org/10.1109/MC.2015.45>.
- VEREVKA, T.; S. GUTMAN a A. SHMATKO, 2019. Prospects for Innovative Development of World Automotive Market in Digital Economy. online. In: *Proceedings of the 2019 International SPBPU Scientific Conference on Innovations in Digital Economy*. s. 1–6. Dostupné z: <https://doi.org/10.1145/3372177.3373320>.
- WU, S., 2023. Analysis of the Core Competitive Advantage of the Auto Industry: A Case of Tesla' s Business Model. online. In: *International Conference on Company Management, Accounting and Marketing*. Dostupné z: <https://doi.org/10.54691/bcpbm.v36i.3494>.