

---

# JAMB - Jednotná aplikace města Brna

---

## Příloha K.1 Smlouvy - Kvalitativní vlastnosti Projektu a Služeb JAMB

### 1. KONVENCE TOHOTO DOKUMENTU

#### 1.1. Obecné

- 1.1.1. Tento dokument specifikuje kvalitativní požadavky na Jednotnou aplikaci města Brna, která zahrnuje mobilní aplikace, backendový systém (včetně integrační vrstvy a administraci), a infrastrukturu nutnou k zajištění provozu celého řešení. Kvalitativní požadavky definují vlastnosti nezbytné pro zajištění škálovatelnosti, spolehlivosti, bezpečnosti a uživatelské přívětivosti systému. Dokument slouží jako základ pro řízení kvality během vývoje a implementace systému a kontrolu při akceptačních testech a v pilotním provozu.
- 1.1.2. Tento dokument je součástí technické části zadávací dokumentace a odkazuje a souvisí s dalšími dokumenty, zejména:
- P - Projekt: Funkční požadavky a popis integrací.
  - S - Služby: Požadavky na provoz, podporu a údržbu a další služby.
  - U - Sledované ukazatele (SLI/SLO): Detailní specifikace ukazatelů a jejich metrik.
- 1.1.3. Všechny body požadavků jsou uvedeny v přítomném či minulém čase a pokud možno bez použití kondicionálů proto, aby každý požadavek šlo vyhodnotit jako splněný či nesplněný jednoduchou odpovědí ano / ne podle jeho aktuálního reálného stavu v době akceptace. Použití minulého času v tomto dokumentu neznamená automatické potvrzení Zákazníka o splnění tohoto bodu. Splnění či nesplnění požadavku je vždy předmětem akceptačního řízení.
- 1.1.4. Pokud z kontextu nevyplývá jinak, slova a slovní spojení v jednotném čísle zahrnují i množné číslo a naopak.

#### 1.2. Pojmy a zkratky

- 1.2.1. Pro potřeby Zadání jsou zkratkou JAMB označeny všechny součásti systému - Aplikace, Backend a Externí služby.
- 1.2.2. Aplikace - označení mobilních aplikací pro iOS a Android.
- 1.2.3. Backend - Technické zázemí systému, zahrnující Integrační vrstvu a Administrace.

- 1.2.4. Integrovaná vrstva - Middleware zajišťující propojení mezi Aplikací a Interními a Externími službami a dále propojení Interních a Externích služeb mezi sebou.
- 1.2.5. Administrace jsou všechny editační, administrační rozhraní a nástroje, které pro Zákazníka vytvořil nebo vytváří Dodavatel.
- 1.2.6. Externí služby jsou všechny služby třetích stran přímo využívaných v JAMB (např. cloudové služby).
- 1.2.7. Interní služby jsou služby města Brna, které jsou integrovány do JAMB (např. autentizační brána, zdroje dat, BrnoID atp.).
- 1.2.8. Platební brána je služba, kterou zajistil Zákazník pro vybírání poplatků a plateb v Aplikaci.
- 1.2.9. Distribuční služby jsou primárně Google Play a Apple App Store jako platformy pro šíření aplikace, případně další distribuční platformy podle dohody Dodavatele se Zákazníkem.
- 1.2.10. Administrátor je uživatel Administrace.
- 1.2.11. Uživatel je uživatel Aplikace. Uživatelé jsou identifikováni pomocí jedinečného ID včetně anonymních Uživatelů.
- 1.2.12. Infrastruktura zahrnuje veškeré technické prostředky a služby, které zajišťují provoz, správu a škálovatelnost JAMB. Jedná se o kombinaci hardwaru (servery, úložiště, sítě), softwaru (operační systémy, middleware, cloudové platformy) a bezpečnostních opatření (firewally, šifrování, zálohování), které společně vytvářejí stabilní, bezpečné a výkonné prostředí pro všechny součásti systému JAMB.
- 1.2.13. MAU (měsíčně aktivní uživatelé) je metrika označující počet unikátních uživatelů, kteří využili aplikaci nebo systém alespoň jednou během daného kalendářního měsíce. MAU představuje počet Uživatelů, kteří během jednoho měsíce aktivně interagovali s Aplikací a využili libovolnou funkcionality JAMB mimo samotného přihlášení. Každý uživatel je počítán pouze jednou za daný měsíc bez ohledu na počet přihlášení nebo aktivit.
- 1.2.14. Za Referenční zařízení jsou považována tři nejběžnější zařízení každé platformy (iOS, Android). Po spuštění projektu se konkrétní Referenční zařízení určují kvartálně podle statistik využívání aplikace. Před uvedením projektu do produkčního provozu jsou Referenční zařízení určena ve fázi analýz.
- 1.2.15. Dokumentační služba je systém pro ukládání a správu dokumentace. Provoz Dokumentační služby zajišťuje Zákazník.
- 1.2.16. Pilotní provoz je předběžná fáze nasazení JAMB, během které je systém testován v reálných podmínkách s omezeným počtem Uživatelů (do 1000 MAU). Cílem je ověřit funkčnost, stabilitu, uživatelskou přívětivost a kompatibilitu aplikace před jejím celoplošným spuštěním.

- 1.2.17. Produkční provoz je plnohodnotné nasazení JAMB, dostupné všem Uživatelům. Tato fáze následuje po úspěšném dokončení Pilotního provozu a zahrnuje kompletní využití všech funkcí Aplikací, Integrací a Administrací.
- 1.2.18. HotLine je služba poskytování podpory Uživatelů JAMB.
- 1.2.19. HelpDesk je programové vybavení Dodavatele a služba, prostřednictvím které poskytuje podporu Zákazníka. Nemusí se jednat o stejný nástroj jako je použit u HotLine.
- 1.2.20. Zkratkou SLI (Service Level Indicator) se rozumí sledované metriky JAMB nebo Služeb, které Zákazníkovi poskytuje Dodavatel.
- 1.2.21. Zkratkou SLO (Service Level Objective) se rozumí konkrétní hodnoty (cíle) nebo rozsahy hodnot jednotlivých SLI. Pro jedno SLI může být více SLO (nedodržení cíle může znamenat různý typ incidentu).

## OBSAH DOKUMENTU

<b>1. KONVENCE TOHOTO DOKUMENTU.....</b>	<b>1</b>
1.1. Obecné.....	1
1.2. Pojmy a zkratky.....	1
<b>OBSAH DOKUMENTU.....</b>	<b>4</b>
<b>2. ORGANIZAČNÍ A PRÁVNÍ POŽADAVKY.....</b>	<b>6</b>
2.1. Organizační požadavky.....	6
2.2. Licence.....	6
2.3. Právní a další předpisy.....	7
<b>3. SLEDOVANÉ UKAZATELE.....</b>	<b>7</b>
3.1. Obecné.....	7
<b>4. POŽADAVKY NA DOKUMENTACI.....</b>	<b>8</b>
4.1. Obecné.....	8
4.2. Externí služby.....	8
4.3. Použité knihovny.....	8
4.4. Návrhová a vývojářská dokumentace.....	9
4.5. Dokumentace API Integrovaných vrstev.....	9
4.6. Dokumentace pro testování.....	10
4.7. Provozní dokumentace.....	10
4.8. Bezpečnostní dokumentace.....	11
4.9. Administrátorská dokumentace.....	12
<b>5. ARCHITEKTONICKÉ POŽADAVKY.....</b>	<b>12</b>
5.1. Znovuvyužití stavebních bloků ICT MMB.....	12
5.2. Clover DX.....	12
5.3. Autentizační brána.....	13
<b>6. UDRŽITELNOST.....</b>	<b>13</b>
6.1. Obecné.....	13
<b>7. VÝKON A ŠKÁLOVATELNOST.....</b>	<b>13</b>
7.1. Obecné.....	13
7.2. Integrovaná vrstva a Backend.....	14
7.2.6. Aplikace.....	14
<b>8. BEZPEČNOST.....</b>	<b>14</b>
8.1. Obecné.....	14
8.2. Administrace.....	17
8.3. Backend.....	18
8.4. Aplikace.....	18
<b>9. ODOLNOST.....</b>	<b>19</b>

9.1. Infrastruktura.....	19
9.2. Integrační vrstva.....	20
9.3. Aplikace.....	20
<b>10. AGILITA.....</b>	<b>20</b>
10.1. Obecné.....	20
10.2. Aplikace.....	21
<b>11. MONITOROVATELNOST.....</b>	<b>21</b>
11.1. Obecné.....	21
<b>12. POUŽITELNOST.....</b>	<b>21</b>
12.1. Aplikace.....	21
<b>13. PŘÍSTUPNOST.....</b>	<b>22</b>
13.1. Obecné.....	22
<b>14. KONZISTENTNOST.....</b>	<b>22</b>
14.1. Obecné.....	22
14.2. Aplikace.....	23
<b>15. KOMPATIBILITA A INTEROPERABILITA.....</b>	<b>23</b>
15.1. Integrační vrstva.....	23
15.2. Aplikace.....	24
15.3. Administrace.....	24
<b>16. PRAKTIKY A METODIKY.....</b>	<b>25</b>
16.1. Automatizace.....	25
16.2. Internacionalizace a lokalizace.....	25
16.3. E-maily.....	26
16.4. Další praktiky a metodiky.....	26
16.5. Zakázané praktiky, technologie a komponenty.....	27
<b>17. NASAZOVÁNÍ NOVÝCH VERZÍ.....</b>	<b>27</b>
17.1. Obecné.....	27
17.2. Integrační vrstva.....	28
17.3. Aplikace.....	28

## 2. ORGANIZAČNÍ A PRÁVNÍ POŽADAVKY

### 2.1. Organizační požadavky

- 2.1.1. Všechny komplexní požadavky na systém vychází z provedených detailních analýz požadavků a jsou zpracovány v písemné podobě kterou schvaluje Zákazník.
- 2.1.2. Všechny externí služby, které má Dodavatel záměr použít, jsou písemně schváleny Zákazníkem. Účty k těmto službám jsou vytvořeny na správcovský účet ve vlastnictví Zákazníka a Dodavatel má k těmto službám (pokud je třeba) nasdílen přístup na svůj samostatný účet.
- 2.1.3. Dodavatel je umožňuje Zákazníkovi provádět audity systému a jeho bezpečnosti během vývoje i v průběhu produkčního provozu.

### 2.2. Licence

- 2.2.1. Licence je upravena ve Smlouvě. Zákazník ke všem Výstupům, u kterých je to možné vzhledem k jejich charakteru, obdrží zdrojové soubory včetně dokumentace a komentářů a v případě zdrojových kódů i jejich kompletní historii. Zdrojové kódy zahrnují i veškeré podpůrné soubory, které jsou potřebné pro nasazení a provoz (např. konfigurační soubory, skripty pro nasazení, databázové schémata).
- 2.2.2. Jsou použity pouze takové open-source komponenty, které jsou kompatibilní s licenčními požadavky projektu, a zajistí, že žádná z použitých součástí nevyžaduje zpětné poskytnutí zdrojových kódů Zákazníka (např. copyleft licence).“
- 2.2.3. Počet Administrátorů ani jiných osob a Uživatelů není licenčně omezen ani samostatně zpoplatněn.
- 2.2.4. Veškeré použité součásti nejsou zatíženy licenčními ani jinými podobnými periodickými poplatky.

## 2.3. Právní a další předpisy

- 2.3.1. Dodavatel při vytváření systému a poskytování služeb dodržuje právní předpisy a interní předpisy Zákazníka. Zákazník upozorňuje zejména na:
- a. Nařízení (EU) 2016/679 (GDPR).
  - b. Směrnice (EU) 2002/58/ES (Nařízení o soukromí a elektronických komunikacích) a Nařízení, které ji nahrazuje (ePrivacy).
  - c. Zák. č. 480/2004, zákon o některých službách informační společnosti a o změně některých zákonů, zejména §7.
  - d. Zákon 181/2014 Sb., o kybernetické bezpečnosti, ve znění pozdějších předpisů.
  - e. Vyhláška č. 82/2018 Sb., o kybernetické bezpečnosti, ve znění pozdějších předpisů.
  - f. Směrnice Evropského parlamentu a Rady (EU) 2016/2102 ze dne 26. října 2016 o přístupnosti internetových stránek a mobilních aplikací subjektů veřejného sektoru.
  - g. Zákon č. 99/2019 Sb., o přístupnosti internetových stránek a mobilních aplikací.
  - h. Zákon č. 365/2000 Sb., o informačních systémech veřejné správy (ISVS).
  - i. Vyhláška č. 360/2023 Sb., o dlouhodobém řízení informačních systémů veřejné správy.
  - j. Vyhlášky č. 315/2021 Sb., 316/2021 Sb. a 190/2023 Sb. o bezpečnostních pravidlech a úrovních pro orgány veřejné moci využívající služby poskytovatelů cloud computingu a o některých požadavcích pro zápis do katalogu cloud computingu. Zákazník klasifikoval JAMB jako ISVS s bezpečnostní úrovní "nízká" pro využívání cloud computingu orgány veřejné moci.
- 2.3.2. Jsou striktně dodržovány Google Play Developer Policies a Apple App Store Review Guidelines. Dodavatel má povinnost upozorňovat Zákazníka na nevhodnost pokynů, požadavků či zadání, které tyto pravidla porušují. Dodavatel má odpovědnost za škody způsobené nedodržením těchto zásad v případě, že na toto porušení Zákazníka neupozorní.

## 3. SLEDOVANÉ UKAZATELE

### 3.1. Obecné

- 3.1.1. V samostatném provozním dokumentu U.1 jsou definovány SLI (Service Level Indicators - vyhodnocované metriky) a k nim příslušné SLO (Service Level Objectives - cíle dosahovaných SLI, většinou jako minimální či maximální hodnota, popř. rozsah hodnot - typicky za udaný čas). U SLI, u kterých to dává smysl se určuje region, ze kterého je prováděn test, referenční klient či obdobné údaje.

- 3.1.2. Minimální rozsah a SLI a hodnoty SLO jsou Zákazníkem stanoveny v dokumentu U.0. Dodavatel v rámci realizace Projektu P.1 stanovil přesné hodnoty a případně rozšířil sledované ukazatele a vytvořil dokument U.1.

## 4. POŽADAVKY NA DOKUMENTACI

### 4.1. Obecné

- 4.1.1. Veškerá dokumentace je primárně v češtině, technická dokumentace je přípustná i v angličtině.
- 4.1.2. Všechny dokumenty používají stejnou terminologii.
- 4.1.3. Veškerá dokumentace je tvořena tak, aby podporovala potřeby Zákazníka pro bezpečnostní, technické a další audity a kontroly veřejnými i soukromými společnostmi a orgány.
- 4.1.4. Zákazník má k dispozici uspořádané a přehledné výstupy všech provedených analýz.
- 4.1.5. Veškerá dokumentace je uložena v Dokumentační službě a obsahuje datum poslední aktualizace a autora změny.
- 4.1.6. Dodavatel vytvořil a udržuje samostatný dokument jako katalog všech dále uvedených dokumentů s jejich verzí, odpovědnou osobou za správu a datem poslední aktualizace.

### 4.2. Externí služby

- 4.2.1. Dodavatel udržuje seznam všech Externích služeb využívaných v JAMB, který obsahuje:
- Název služby
  - Odkaz na službu
  - Účel použití služby
  - SLA definované poskytovatelem
  - Bezpečnostní požadavky služby (např. způsob autentizace, šifrování komunikace)
  - Limity a kvóty (např. maximální počet API požadavků za měsíc).

### 4.3. Použité knihovny

- 4.3.1. Dodavatel udržuje seznam všech knihoven využívaných v JAMB, který obsahuje:
- Název knihovny
  - Účel použití knihovny
  - Odkaz na zdrojové kódy
  - Licence



#### 4.4. Návrhová a vývojářská dokumentace

- 4.4.1. JAMB je dokumentován v souladu s Národním architektonickým rámcem (NAR) vycházejícího z mezinárodně uznávaných standardů tvorby a údržby architektury úřadů TOGAF a ArchiMate, spravovaných The Open Group.
- 4.4.2. Dodavatel udržuje katalog funkčních a nefunkčních požadavků JAMB.
- 4.4.3. Dodavatel udržuje prototyp a návrh uživatelského rozhraní (UX/UI) Aplikace.
- 4.4.4. Dodavatel udržuje historii verzí (changelog) Aplikace i Backendu.
- 4.4.5. Dodavatel předal Zákazníkovi vývojářskou dokumentaci v písemné podobě, obsahující minimálně:
  - a. Popis základní logiky/filozofie produktu.
  - b. Popis logické architektury systému, všech jeho komponent a jejich vazeb včetně diagramů.
  - c. Popis infrastruktury včetně deployment diagramu.
  - d. Dokumentace návrhu databáze.
  - e. Popis klíčových aplikačních entit a jejich vztahů.
  - f. Popis mechanismu lokalizace včetně workflow pro aktualizace lokalizací.
  - g. Definice coding standards.
  - h. Popis release procesu (build a deployment do všech prostředí).
  - i. Popis verzovacího workflow.
  - j. Popis deployment procesu; slovně a pomocí diagramu, z něhož budou patrné jednotlivé stavy a operace během vývoje a nasazování aplikace.

#### 4.5. Dokumentace API Integrovaní vrstvy

- 4.5.1. Dodavatel předal Zákazníkovi a udržuje aktuální kompletní dokumentaci API Integrovaní vrstvy. Tato dokumentace je vytvořena ve standardizovaném formátu, který je kompatibilní s nástroji jako OpenAPI (Swagger), RAML nebo API Blueprint. Preferovaným formátem je OpenAPI ve verzi 3.1, pokud není Zákazníkem schválen jiný formát. Dokumentace musí být vhodná pro automatizované zpracování a snadno integrovatelná do nástrojů třetích stran pro testování, modelování a vizualizaci API. Součástí dokumentace jsou verze jednotlivých API, aby bylo možné zajistit konzistenci a zpětnou kompatibilitu. Dokumentace API obsahuje minimálně:
  - a. Přehled všech dostupných endpointů API.
  - b. Metody a parametry jednotlivých endpointů (vstupní data, výstupy, chybové stavy).

- c. Autentizační mechanismy a požadavky na zabezpečení (např. OAuth2, API klíče, certifikáty).
- d. Struktury a příklady požadavků a odpovědí (např. ve formátu JSON nebo XML).
- e. Specifikace limitů a chybových stavů API.

## 4.6. Dokumentace pro testování

- 4.6.1. Dodavatel vytváří, spravuje a udržuje testovací scénáře.
- 4.6.2. Součástí dokumentace je popis používaných testovacích nástrojů a způsobu zajištění testovací automatizace a popis způsobu testování integrací s externími a interními službami.

## 4.7. Provozní dokumentace

- 4.7.1. Dodavatel předal Zákazníkovi dokumentaci v písemné podobě, obsahující minimálně:
  - a. Dokumentace kompletní infrastruktury a repository s šablonami pro automatické nastavení infrastruktury.
  - b. Popis škálovatelnosti aplikace a infrastruktury, včetně automatických škálovacích mechanismů. Popis limitů systému a postupy pro přidávání dalších zdrojů (např. vertikální a horizontální škálování infrastruktury).
  - c. Detailní krokový instalační manuál pro Backend včetně přesných požadavků na infrastrukturu (HW i SW) pro vývojové i produkční prostředí.
  - d. Popis případných změn v nastavení operačních systémů.
  - e. Popis konfigurace aplikačních a webových serverů a konfigurací databází.
  - f. Seznam externích služeb, závislostí a datových toků (např. Mailchimp, Sentry, DataDog, CRM, ERP apod.)
  - g. Dokumentace periodických procesů (typicky cron jobs).
  - h. Dokumentace k používaným automatizacím (hooks, makefiles, playbooks, ...).
  - i. Dokumentace typů zasílaných e-mailů a způsobu jejich posílání (SMTP servery či služby a jejich požadavky na DNS záznamy).
  - j. Seznam standardních provozních úkonů a pracovních postupů pro správu systému.
  - k. Dokumentace k integracím či importům dat z externích zdrojů.
  - l. Dokumentace k logování včetně formátu logů, typu událostí, které jsou logovány, strategie ukládání a retenčních pravidel pro logy. Popis mechanismu zabezpečení logů (např. přístupová práva, šifrování).

- m. Popis mechanismů push notifikací, způsob správy uživatelských opt-in preferencí a konfigurace notifikačních služeb.
- n. Detailní popis řešení zálohování a obnovy, včetně kompletních postupů Disaster Recovery. Dodavatel vytvořil a dále udržuje stále aktuální dokumentaci jednoznačně upravující kroky vedoucí k zajištění plné obnovy systému po havárii mající globální dopad na chod systému s ohledem na minimalizaci škod. Dokumentace DRP (Disaster Recovery Plan) je zpracována do nejmenšího detailu, to znamená vytvoření detailního postupu obnovy každé komponenty včetně popisu všech kroků vedoucí k její obnově. Pravidla zálohování obsahují vždy alespoň:
  - specifikaci zálohovaných komponent (co je nutné zálohovat?)
  - způsob jejich zálohování včetně časové návaznosti jednotlivých komponent (jakým způsobem se záloha má realizovat?)
  - periodu zálohování (kdy / jak často se má záloha provádět?)
  - retenční pravidla pro dobu a počet verzí uchovávaných záloh (jak dlouho a kolik verzí záloh se má uchovávat?)
- o. Seznam administrátorských a servisních účtů k použitým operačním systémům, aplikacím a databázím.
- p. Popis nastavení monitoringu a dohledu včetně použitých notifikací a jejich konfigurace.
- q. Dokumentace vyplývající ze zákona č. 365/2000 Sb. o ISVS, v platném znění, včetně souvisejících předpisů (zejména vyhláška č. 360/2023 Sb. o dlouhodobém řízení informačních systémů veřejné správy) pokud již není součástí jiné dokumentace podle tohoto zadání.

## 4.8. Bezpečnostní dokumentace

- 4.8.1. Dodavatel předal Zákazníkovi bezpečnostní dokumentaci v písemné podobě, minimálně v rozsahu:
  - a. Politika práce s hesly, klíči a certifikáty a způsob a místa jejich ukládání.
  - b. Popis mechanismů detekce anomálií a jejich reportování (např. IDS/IPS).
  - c. Seznam osobních údajů, se kterými systém pracuje včetně kategorie.
  - d. Diagram, kudy putují osobní údaje systémem a kde jsou uložena.
  - e. Seznam třetích stran, které mají přístup k osobním datům a (odkaz na) smlouvy s těmito stranami
  - f. Popis použitých kryptografických prostředků, protokolů a nástrojů zejména pro účely auditů.

- g. Pokud je používána infrastruktura Zákazníka, ke které je třeba, aby Zákazník zřídil síťové prostupy, tabulka požadovaných síťových prostupů, ke každé povolené komunikaci obsahuje alespoň:
  - Zdrojová adresa / Skupina adres
  - Cílová adresa / Skupina adres
  - Cílový komunikační port / Skupina komunikačních portů
  - Poznámka (stručný název důvodu zřízení prostupu)
- h. Seznam všech použitých TLS certifikátů s dobou platnosti na kterou jsou vydávány včetně popisu a postupu pro jejich obnovu.
- i. Součástí bezpečnostní dokumentace je postup při řešení bezpečnostních incidentů, včetně jejich evidence a reportování.

#### 4.9. Administrátorská dokumentace

- 4.9.1. Existuje administrátorská dokumentace - návod na zadávání a úpravu obsahu ve všech Administracích. Dokumentace předpokládá základní dovednosti ovládání PC a intranetových aplikací. Zaměřuje se zejména na specifika JAMB a jednotlivé operace a postupy v něm prováděné. V dokumentaci jsou zachyceny významné důsledky a návaznosti jednotlivých operací.
- 4.9.2. Administrátorská dokumentace je členěná podle uživatelských rolí v systému a obsahuje i popis přiřazování a správy oprávnění na základě rolí.
- 4.9.3. Průběžně aktualizované způsoby řešení častých problémů a doporučených postupů pro jejich řešení.

### 5. ARCHITEKTONICKÉ POŽADAVKY

#### 5.1. Znovuvyužití stavebních bloků ICT MMB

Smysluplně se využívají stavební bloky ICT MMB a standardizované komponenty, zejména Clover DX a Autentizační brána. Dále např. ISRZ proxy a NIA proxy.

#### 5.2. Clover DX

Je použita Zákazníková ETL Platforma Clover DX.

Jedná se o platformu, která zajišťuje základní funkcionalitu celé řady služeb poskytovaných Magistrátem města Brna. Integrační Datová Platforma je využívána pro komunikaci mezi různými IS Magistrátu města Brna. Komunikace mezi těmito IS probíhá na základě požadavku, buď 1:1 nebo tato platforma umožňuje transformaci dat vyplývající z možností komunikujících systémů.

Rozsah služeb spojených s nastavením platformy Clover DX není součástí této VZ.

### 5.3. Autentizační brána

Je použita Autentizační brána MMB, která podporuje autentizační protokoly pro napojení SP a IdP SAML 2.0, OpenID Connect.

## 6. UDRŽITELNOST

### 6.1. Obecné

- 6.1.1. Dodavatel vytváří JAMB a každou jeho část tak, aby jejich další změny či rozvoj mohl po skončení Smlouvy realizovat kterýkoliv jiný odborník v oboru. To znamená, že Dodavatel:
  - a. vytváří zdrojové kódy, které budou předány Zákazníkovi, podle nejnovějších a nejlepších standardů a norem, v přehledné a strukturované formě a včetně přehledných a srozumitelných komentářů;
  - b. ke všem funkcionalitám JAMB vede písemnou Dokumentaci, kde budou popsány jednotlivé funkce, logické a technické vazby mezi nimi, vysvětlivky a další potřebné informace, aby mohl na rozvoj JAMB či jeho změny navázat bez obtíží jiný odborník v oboru;
  - c. je odborně, kapacitně i personálně připraven po skončení smlouvy poskytnout součinnost a za stejných cenových podmínek jako při poskytování servisních služeb předat JAMB další straně, kterou určí Zákazník.
- 6.1.2. Všechny použité technologie (frameworky, knihovny, služby) budou podporované jejich výrobcí alespoň po dobu 5 let. V případě předčasného ukončení podpory je zajištěna kvalitativně srovnatelná náhrada technologie.
- 6.1.3. Backend i Aplikace jsou rozděleny do modulárních komponent s jasně definovanými a dokumentovanými rozhraními.
- 6.1.4. JAMB je navržen tak, aby provozní náklady byly optimalizované a dlouhodobě udržitelné.

## 7. VÝKON A ŠKÁLOVATELNOST

### 7.1. Obecné

- 7.1.1. Všechny komponenty JAMB musí být optimalizovány pro vysoký výkon (např. minimalizace výpočtů na straně klienta, využití caching mechanismů).
- 7.1.2. Všechny součásti JAMB využívají caching mechanismy pro snížení zatížení Backendu (zejména Integrovaná vrstva) a zlepšení rychlosti odezvy.
- 7.1.3. Pro velká a/nebo nepersonalizovaná data (např. obrázky a data, která se mění méně často) se využívá CDN.
- 7.1.4. JAMB je připraven na zátěžové testy (load testing, stress/spike testing, scalability testing).
- 7.1.5. JAMB implementuje mechanismy, které minimalizují dopad výpadku externích služeb na celkový výkon systému (např. fallback mechanismy, retry strategie).

## 7.2. Integrační vrstva a Backend

- 7.2.1. Integrační vrstva implementuje správu prostředků (např. connection pooling) pro efektivní využití databází a externích API.
- 7.2.2. Integrační vrstva efektivně spravuje uživatelské relace tak, aby byl minimalizován dopad na výkon při velkém počtu současných uživatelů.
- 7.2.3. Integrační vrstva je schopna horizontálního škálování (např. přidání dalších serverů nebo instancí) pro zvýšení kapacity.
- 7.2.4. Návrh API pro Aplikace minimalizuje velikost přenášených dat pomocí komprese a optimalizovaných datových struktur.
- 7.2.5. Backend je schopen vertikálního škálování, přidání dalších zdrojů (např. více CPU, paměti) pro zvýšení výkonu jednotlivých komponent.

### 7.2.6. Aplikace

- 7.2.7. Aplikace je optimalizována tak, aby minimalizovala spotřebu baterie, CPU a paměti.
- 7.2.8. Pro snížení zátěže backendu a zajištění uživatelského komfortu aplikace podporuje ukládání často používaných dat do lokální cache s možností offline přístupu.
- 7.2.9. Aplikace efektivně provádí synchronizaci dat při návratu z offline režimu, aniž by došlo k nadměrnému zatížení backendu.
- 7.2.10. Aplikace implementuje retry mechanismy s exponenciálním zpožděním pro opakované neúspěšné požadavky.
- 7.2.11. Aplikace má zabudované mechanismy pro sledování klíčových metrik výkonu, jako je rychlost načítání, doba odezvy, počet chybových stavů a vytížení zařízení a je integrovaná s nástroji pro sledování chyb (např. Firebase Crashlytics).
- 7.2.12. Aktualizace aplikace je možné nasazovat postupně, aby bylo možné testovat jejich vliv na výkon.
- 7.2.13. Při použití webview se používá podpora komprese obsahu, HTTP/2, asynchronní načítání externích zdrojů a lokální caching.

## 8. BEZPEČNOST

### 8.1. Obecné

- 8.1.1. JAMB netrpí základními zranitelnostmi, zejména
  - Aktuální OWASP Top 10 a OWASP Mobile Top 10
  - Obcházení autorizace - např. přístup k datům jiných zákazníků/uživatelů nebo funkcím správce z běžného účtu

- Nezabezpečené session ID - např. token, který lze uhodnout; token uložený na nezabezpečeném místě atp.
- Injections - SQLi, NoSQLi, XXE, OS command injection, ...
- Cross-site scripting (XSS) - např. volání nezabezpečených funkcí JavaScriptu, provádění nezabezpečených manipulací s DOM, výpis uživatelského vstupu do HTML bez escapování.
- Cross-site request forgery (CSRF) - např. zpracování požadavků s hlavičkou Origin z jiné domény.
- Použití knihoven se známými zranitelnostmi
- Další zranitelnosti, které je možno detekovat běžnými automatizovanými nástroji

8.1.2. Nepoužívají se výchozí přihlašovací údaje.

8.1.3. Je dodržován princip nejmenšího oprávnění (least privilege).

8.1.4. Nejsou veřejně přístupné interní a vývojové soubory a adresáře jako např. .git repozitář, konfigurační soubory pro vývoj, sestavení nebo provoz, source maps atp.

8.1.5. Jako zdroj aktuálních best practices je považován <https://cheatsheetseries.owasp.org>

8.1.6. Přístup k citlivým datům (osobní a přístupové údaje) je omezen výhradně na pracovníky Dodavatele s oprávněnou potřebou.

8.1.7. Oprávněná potřeba přístupu k citlivým datům či backendu je pravidelně kontrolována. Nadbytečné účty či přístupová oprávnění pracovníků Dodavatele bez oprávněné potřeby jsou bez zbytečného prodlení zrušeny.

8.1.8. Je zajištěno, že neprodukční prostředí neobsahují produkční citlivá data. Dodavatel nekopíruje či nepřesouvá citlivá data z produkčního prostředí Zákazníka, pokud to Zákazník výslovně neschválil. Osobní data jsou při kopírování z produkčního prostředí vždy anonymizována.

8.1.9. V případech, kde systém zajišťuje autentizaci uživatelů, tak práce s hesly respektuje:

- a. minimální délka hesla je 8 znaků pro běžné a 12 znaků pro administrátorské účty
- b. maximální délka hesla není omezena na méně než 64 znaků
- c. nejsou omezeny povolené znaky, které lze použít
- d. nepoužívají se tajné otázky jako jediný požadavek na obnovení hesla
- e. při změně hesla se vyžaduje aktuální heslo a e-mailové ověření změny
- f. nově vytvořená hesla jsou

- ověřována podle seznamů běžných hesel
  - algoritmicky kontrolována, že neobsahují opakování typu aaaa nebo sekvence typu 1234
  - algoritmicky kontrolána, že v heslu není část e-mailu nebo jména Uživatele či brandu a používaných značek Zákazníka
- g. nově vytvořená hesla jsou ověřována podle databází uniklých hesel
- h. hesla jsou ukládána v hashovaném a salted formátu za použití paměťově nebo výpočetně náročné jednosměrné hashovací funkce dle aktuálního doporučení NÚKIB v oblasti kryptografických prostředků
- i. při detekci útoku pomocí hrubé síly je vynuceno vhodné uzamčení / ochrana proti přístupu k účtu
- 8.1.10. Ve Version Control System (VCS) nejsou uloženy žádná privátní hesla, certifikáty, klíče, přístupové údaje atp. (secrets). Výjimku tvoří secrets, které jsou společně s ostatními konfiguračními parametry uloženy v samostatném repozitáři a šifrovány bezpečným způsobem.
- 8.1.11. V případě, že použitá součást obsahuje bezpečnostní chybu střední a větší závažnosti relevantní k systému, je součást aktualizována nejpozději do 30 kalendářních dnů, pokud je splněno:
- a. Chyba má přidělený CVE identifikátor a současně
  - b. Existuje opravná verze či workaround od Dodavatele či autora této součásti
  - c. Nedošlo k písemné dohodě o tom, že se chyba nebude řešit
- 8.1.12. V případě, že použitá součást dle předchozího bodu obsahuje bezpečnostní chybu, které bylo přiděleno CVSS 3.x skóre  $\geq 7$ , musí být použitá součást aktualizována nejpozději
- a. do tří dnů pro CVSS 3.x skóre  $\geq 9$ ,
  - b. do sedmi dnů pro CVSS 3.x skóre  $\geq 8$ ,
  - c. do deseti dnů pro CVSS 3x skóre  $\geq 7$ .
- Jako počátek lhůty je považováno datum zápisu bezpečnostní chyby do databáze NIST NVD (NVD Published Date). Po uplynutí lhůty se nutnost aktualizace považuje za incident kategorie 2.
- 8.1.13. Je použit serverový certifikát schválený Zákazníkem. Jeho nasazování je automatizováno a platnost automaticky monitorována. Není použit certifikát s platností delší než 12 měsíců, klíč certifikátu se rotuje minimálně jednou ročně.



- 8.1.14. V JAMB jsou implementovány všechny relevantní funkcionality, které vyžaduje GDPR s ohledem na zpracovávané osobní údaje. Výjimku tvoří operace, které manuálně provede Dodavatel v rámci poskytování Podpory.
- 8.1.15. Všechna data přenášená mezi Aplikací, Backendem, interními a externími systémy jsou šifrována pomocí protokolu TLS 1.2 nebo vyššího.
- 8.1.16. Citlivá data uložená v Backendu nebo Aplikaci (např. osobní údaje, přístupové tokeny) jsou šifrována pomocí algoritmů splňující aktuální požadavky NÚKIB<sup>1</sup>.
- 8.1.17. JAMB používá moderní a bezpečné autentizační mechanismy (např. OAuth 2.0, JWT, API klíče).
- 8.1.18. Přístup ke všem administrativním funkcím a API je řízen na základě uživatelských rolí (RBAC).
- 8.1.19. Všechny vstupy od Uživatelů jsou validovány, sanitizovány a kontrolovány proti injekčním a dalším útokům.
- 8.1.20. JAMB nepoužívá zastaralé algoritmy (např. MD5, SHA-1) pro šifrování nebo hashování.
- 8.1.21. Jsou používány pouze důvěryhodné a ověřené knihovny nebo komponenty třetích stran.

## 8.2. Administrace

- 8.2.1. Uživatelé administrací mají k dispozici možnost aktivovat MFA (multi factor authentication, vícefaktorové ověřování). Pro role určené Zákazníkem je použití MFA povinné.
- 8.2.2. Přístupy k citlivým datům a administrátorské akce, které jsou logovány jsou chráněny před manipulací.
- 8.2.3. Neexistují společné přístupové účty, každý pracovník Dodavatele i Zákazníka má samostatný přístup vedený na jeho jméno.
- 8.2.4. U veřejně přístupných administrací je nasazen soubor security.txt podle posledního Internet-Draft nebo RFC.
- 8.2.5. V URL není nikdy osobní údaj.
- 8.2.6. Na stránkách obsahujících osobní údaje není použit JavaScript načítaný od třetích stran, pokud není explicitně schválen Zákazníkem.
- 8.2.7. Inline JavaScript (script type="text/javascript") je povolen pouze s nonce atributem, který se mění pro každý Request. Inline application/json je povolený i bez nonce.

---

<sup>1</sup> Minimální požadavky na kryptografické algoritmy  
[https://nukib.gov.cz/download/uredni\\_deska/Minimalni%20požadavky%20na%20kryptograficke%20algoritmy.pdf](https://nukib.gov.cz/download/uredni_deska/Minimalni%20požadavky%20na%20kryptograficke%20algoritmy.pdf)

### 8.3. Backend

- 8.3.1. Backendová API obsahují mechanismy omezující počet požadavků z jedné IP adresy (rate limiting).
- 8.3.2. Backend implementuje mechanismy pro detekci a prevenci opětovného použití zachycených dat (např. nonce, časové značky).
- 8.3.3. Backend je chráněn pomocí firewallu, který monitoruje a blokuje podezřelé aktivity.
- 8.3.4. Backend je navržen a vytvořen v souladu s bezpečnostními standardy OWASP Top 10.
- 8.3.5. Pro všechna HTTPS URL je posílána Strict Transport Security hlavička.
- 8.3.6. Všechny cookie mají nastavený příznak Secure.
- 8.3.7. Session cookie mají nastavené příznaky HttpOnly a SameSite.
- 8.3.8. Významné akce, zejména v Administracích, obsahují CSRF tokeny.
- 8.3.9. Používají se bezpečnostní hlavičky X-Frame-Options, X-Content-Type-Options, Referrer-Policy a Permissions-Policy.
- 8.3.10. Je definována bezpečná Content Security Policy (CSP).
- 8.3.11. Stránky při přístupu přes protokol HTTP korektně (tj. se zachováním FQDN) přesměrovávají na stejné URL s protokolem HTTPS.
- 8.3.12. Obsah a funkce jsou dostupné pouze pomocí protokolu HTTPS, přístup pomocí HTTP protokolu je umožněn pouze pro přesměrování na zabezpečenou variantu příslušného zdroje.
- 8.3.13. Všechny zdroje vkládané z jiných serverů, včetně iframes, jsou vloženy výhradně za použití protokolu HTTPS.
- 8.3.14. Načítání assetů (javascript, CSS, fonty, obrázky atp.) ze serverů třetí strany musí být vždy schváleno Zákazníkem. V těchto případech je vždy použito SRI (Subresource Integrity), pokud to server třetí strany podporuje.
- 8.3.15. Je zajištěno bezpečné oddělení přístupu Uživatelů aplikace od interních systémů a Interních služeb MMB.

### 8.4. Aplikace

- 8.4.1. Aplikace je distribuována pouze prostřednictvím Distribučních služeb a je podepsána ověřenými certifikáty.
- 8.4.2. Aplikace vyžaduje pouze nezbytná oprávnění a poskytuje uživateli kontrolu nad jejich udělením.
- 8.4.3. Aplikace podporuje automatické aktualizace a informování uživatele o nutnosti instalace kritických oprav.

- 8.4.4. Osobní, citlivá nebo důvěrná data jsou ukládána do zabezpečených úložišť, např. Keychain na iOS nebo Secure Storage na Androidu.
- 8.4.5. Aplikace implementuje mechanismy, které minimalizují útoky typu MITM (např. certificate pinning).
- 8.4.6. Zdrojový kód aplikace je obfuskován (např. pomocí ProGuard pro Android a SwiftShield pro iOS), aby bylo sníženo riziko reverzního inženýrství. Produkční verze aplikací není možné debuggovat.
- 8.4.7. Aplikace obsahuje mechanismy pro detekci a ochranu proti neoprávněným úpravám (např. kontrola integrity podpisu aplikace).
- 8.4.8. Aplikace podporuje využití biometrické autentizace (např. Face ID, Touch ID), pokud to zařízení podporuje.
- 8.4.9. Aplikace je navržena a vytvořena v souladu s bezpečnostními standardy OWASP Mobile Top 10.
- 8.4.10. Při použití webview se zpracovává obsah pouze z předem definovaných URL a není dovoleno načítání obsahu z neověřených zdrojů.
- 8.4.11. Pokud obsah webview nevyžaduje spouštění JavaScriptu, je jeho podpora zakázána.
- 8.4.12. Citlivé údaje nejsou nikdy zasílány nezabezpečenými kanály (e-mail, SMS, notifikace OS, atp.). Hesla nejsou zasílána nikdy, pokud fungují na více než jedno použití.
- 8.4.13. Data Aplikací jsou vyloučena ze zálohování.
- 8.4.14. Je bezpečně nastavena izolace Aplikací (IAC)
  - na platformě Android Jsou exportovány pouze ty aktivity (activity), poskytovatelé obsahu (content providers), služby (service) a broadcast receivers, které je nezbytné exportovat pro správnou funkci aplikace. Exportované komponenty jsou zabezpečené pomocí mechanismů ověřování a oprávnění.
  - na platformě iOS se nepoužívají App Groups a jsou ověřovány příchozí požadavky z URL schemes a Universal Links na důvěryhodnost zdroje (kontrola zdrojové aplikace, validace parametrů).

## 9. ODOLNOST

### 9.1. Infrastruktura

- 9.1.1. Kritické komponenty Infrastruktury jsou replikovány v minimálně dvou geograficky oddělených lokalitách.
- 9.1.2. Data jsou synchronizována v reálném čase mezi primárním a záložním datovým centrem
- 9.1.3. Je možná obnova dat ke specifickému časovému bodu (point-in-time recovery).

## 9.2. Integrační vrstva

- 9.2.1. Komunikace s externími i interními službami implementuje mechanismus pro opakování neúspěšných požadavků s exponenciálním zpožděním.

## 9.3. Aplikace

- 9.3.1. Aplikace ukládá offline data do lokální cache a synchronizuje je s backendem po obnovení připojení.
- 9.3.2. Při problémech s načítáním obsahu do webview (např. výpadek serveru, neexistující stránka) Aplikace zobrazuje uživatelsky přívětivé chybové hlášení místo standardního chybového zobrazení prohlížeče.
- 9.3.3. Aplikace zůstává funkční a stabilní za běžného i zvýšeného provozu a vykazuje konzistentní chování v souladu s definovanými metrikami spolehlivosti.
- 9.3.4. Všechny části aplikace zvládnou výjimečné situace (např. výpadek internetu, nedostupnost backendu, nekompletní nebo nevalidní data API odpovědí) bez pádu aplikace nebo narušení dat.

# 10. AGILITA

## 10.1. Obecné

- 10.1.1. Všechny komponenty systému jsou navrženy modulárně, s jasně definovanými rozhraními, aby změny v jedné komponentě neovlivnily ostatní.
- 10.1.2. Nové funkce mohou být přidány do systému bez nutnosti zásadních změn existujícího kódu.
- 10.1.3. Backend a Aplikace mohou být nasazovány a aktualizovány nezávisle.
- 10.1.4. CI/CD pipeline umožňuje automatické nasazování do testovacího prostředí po každém zápisu změn v kódu do repozitáře.
- 10.1.5. Všechny významné parametry jsou nastavitelné prostřednictvím konfiguračních souborů nebo proměnných prostředí.
- 10.1.6. Využívají se feature-flags pro Aplikace i Integrační vrstvu.
- 10.1.7. Feature flags jsou implementovány tak, aby změny jejich hodnot byly okamžitě účinné bez potřeby restartu.
- 10.1.8. Feature flags umožňují aktivaci/deaktivaci funkcí na úrovni:
- Globální (funkce je dostupná pro všechny uživatele)
  - Role-based (funkce je dostupná pouze pro specifické uživatelské role)
  - User-specific (funkce je dostupná pro konkrétní uživatele)

## 10.2. Aplikace

- 10.2.1. Aplikace umožňuje snadnou změnu nastavení webview (např. přidání nových povolených URL adres).

## 11. MONITOROVATELNOST

### 11.1. Obecné

- 11.1.1. JAMB shromažďuje metriky o výkonu (např. latence API, využití CPU, paměti) pro všechny klíčové komponenty.
- 11.1.2. Metriky jsou ukládány ve standardizovaném formátu a dostupné prostřednictvím monitorovací platformy.
- 11.1.3. Aplikace odesílá klíčové uživatelské metriky (např. doba načítání, počet pádů) do centralizovaného systému pro analýzu.
- 11.1.4. Backendová API exportují metriky včetně počtu úspěšných a neúspěšných požadavků, doby zpracování a počtu aktivních relací.
- 11.1.5. Všechny komponenty systému logují události včetně:
- Chyby a varování
  - Důležité systémové události (např. nasazení nových verzí).
  - Významné akce uživatelů (např. přihlášení, změna hesla, nákupy, změny provedené administrátory).
- 11.1.6. Logy jsou strukturované (např. ve formátu JSON) a zahrnují časové razítko, úroveň závažnosti, zdroj a kontext události.
- 11.1.7. Logy jsou centralizovány do logovacího systému.
- 11.1.8. Systém implementuje distribuované trasování pro sledování průchodu požadavků mezi komponentami (např. aplikace ↔ backend ↔ externí služby). Trasy zahrnují unikátní identifikátory, které umožňují sledovat konkrétní požadavky a jejich související operace.
- 11.1.9. Logy obsahují unikátní ID každé operace (pro sledování požadavků) a záznam výjimek zahrnuje stack trace.

## 12. POUŽITELNOST

### 12.1. Aplikace

- 12.1.1. Jsou dodržovány Apple Human Interface Guidelines a Google Material Design Guidelines.
- 12.1.2. Rozhraní aplikace odpovídá principům jednoduchosti a konzistence (např. stejný design pro podobné akce a navigaci).

- 12.1.3. Použitá terminologie je srozumitelná cílovým uživatelům a reflektuje běžný jazyk (např. „Zaplatit parkování“ místo technických termínů).
- 12.1.4. Interaktivní prvky (např. tlačítka, vstupní pole) jsou jasně identifikovatelné a dostupné pro uživatele bez nutnosti učení.
- 12.1.5. Webview musí podporovat gesta používaná v nativní aplikaci (např. swipe pro návrat zpět).

## 13. PŘÍSTUPNOST

### 13.1. Obecné

- 13.1.1. Jsou respektována Web Content Accessibility Guidelines 2.1 minimálně v úrovni shody AA.
- 13.1.2. Jsou dodržovány Apple Accessibility Guidelines a Android Accessibility Guidelines.
- 13.1.3. Aplikace a její výstupy jsou přístupné pro uživatele s libovolným typem postižení (např. postižení zraku, sluchu, pohybu a motoriky, specifické poruchy učení, psychické a neurologické onemocnění).
- 13.1.4. Aplikace je plně kompatibilní s čtečkami obrazovky, jako je VoiceOver (iOS) a TalkBack (Android).
- 13.1.5. Navigace v aplikaci umožňuje použití gest čteček obrazovky (např. přejetí prstem pro přechod mezi prvky).
- 13.1.6. Všechny interaktivní prvky (tlačítka, odkazy, vstupní pole) obsahují alternativní text nebo popisek.
- 13.1.7. Aplikace respektuje nastavení uživatele na systémové úrovni (např. Dark Mode, High Contrast Mode).
- 13.1.8. Barvy nejsou používány jako jediný indikátor stavu nebo významu (např. chyba musí být doplněna ikonou nebo textem, nejen červenou barvou).
- 13.1.9. Aplikace podporuje dynamické škálování textu podle nastavení zařízení (např. větší text pro slabozraké uživatele).
- 13.1.10. Rozhraní zůstává funkční a v rámci možností přehledné i při maximální velikosti textu.
- 13.1.11. Aplikace respektuje nastavení „omezené pohybové animace“ v systému (např. Reduce Motion na iOS).
- 13.1.12. Aplikace je při vývoji testována pomocí nástrojů jako Accessibility Scanner (Android), Xcode Accessibility Inspector (iOS), nebo jiných automatizovaných nástrojů.

## 14. KONZISTENTNOST

### 14.1. Obecné

- 14.1.1. JAMB zajišťuje, že data jsou konzistentní mezi všemi jeho částmi (např. mezi mobilní aplikací, backendem a databází).

14.1.2. Všechny části systému přistupují k datům přes jednotnou API vrstvu, aby byla zajištěna konzistence dat.

14.1.3. Funkce, které jsou sdílené mezi více platformami, jsou implementovány jednotně a centralizovaně (v ideálním případě na backendu).

## 14.2. Aplikace

14.2.1. Je dodržována konzistentnost uživatelského rozhraní

- Všechny obrazovky aplikace využívají stejné principy navigace (např. tab bar, hamburger menu).
- Položky menu jsou organizovány logicky a konzistentně napříč celým systémem.
- Rozhraní dodržuje jednotné barvy, typografii a ikonografii
- Texty (včetně chybových hlášek a notifikací) používají konzistentní tón a styl komunikace.
- Akce, jako je kliknutí na tlačítko nebo posunutí obrazovky, se chovají stejně na všech obrazovkách aplikace.
- Všechny chybové hlášky a potvrzení akce mají jednotný vzhled a strukturu.
- Aplikace na iOS i Androidu respektují nativní konvence a doporučení příslušné platformy

## 15. KOMPATIBILITA A INTEROPERABILITA

### 15.1. Integrační vrstva

15.1.1. Je využívána aplikační sběrnice Clover DX.

15.1.2. Integrace interních služeb je navržena modulárně, aby umožnila snadné přidání nových funkcionalit nebo zdrojů dat.

15.1.3. API má definovaný a verzovaný kontrakt podle standardu OpenAPI nebo GraphQL Schema Definition Language (SDL).

15.1.4. Všechny změny API jsou v maximální možné míře zpětně kompatibilní. Změny v nové verzi nesmí rozbít stávající aplikace. Pokud není možné zpětnou kompatibilitu zajistit, je vytvořena nová verze API. Plně je podporována alespoň jedna předchozí verze API, aby Aplikace nevyžadovala okamžitou aktualizaci.

15.1.5. Automaticky generovaná dokumentace API (např. Swagger UI, Postman Collection) je dostupná pro všechny relevantní uživatele.

15.1.6. API implementuje validaci vstupních a výstupních dat.

15.1.7. Každý endpoint obsahuje detailní popis, vstupy, výstupy, příklady a možná chybová hlášení.

- 15.1.8. API endpointy, parametry a datové struktury používají jednotné názvosloví.
- 15.1.9. HTTP metody jsou používány korektně s ohledem na jejich idempotence / safety.
- 15.1.10. Všechny odpovědi API obsahují konzistentní formát (např. JSON), včetně struktury pro chybové odpovědi.
- 15.1.11. Kde je relevantní tak API podporuje stránkování (pagination) a filtrování datových sad, aby nedocházelo k přenosu velkých objemů dat.
- 15.1.12. API odpovědi, které jsou často dotazované a málo se mění, umožňují využití cachingu pomocí hlaviček HTTP (např. Cache-Control).
- 15.1.13. API umožňuje klientům specifikovat pouze data, která jsou potřebná (např. pomocí GraphQL nebo dotazových parametrů typu fields).
- 15.1.14. HTTP API vrací smysluplné a konzistentní chybové kódy odpovídající specifikaci HTTP.
- 15.1.15. API systému je navrženo tak, aby podporovalo snadné přidávání nových endpointů bez ovlivnění stávajících funkcí.
- 15.1.16. API je pokryto unit a integračními testy.

## 15.2. Aplikace

- 15.2.1. Aplikace je kompatibilní s:
  - Android ve verzích 10 a vyšší
  - iOS ve verzích 14 a vyšší.
- 15.2.2. Aplikace plně využívá nativní prvky a doporučené vývojářské frameworky platformy.
- 15.2.3. Funkce aplikace zohledňují rozdíly mezi platformami (např. způsoby navigace, notifikace) a přizpůsobují se jejich standardům.
- 15.2.4. Aplikace je testována minimálně na Referenčních zařízeních.
- 15.2.5. Aplikace nemusí být optimalizovaná pro tablety, ale musí na nich fungovat obdobně jako na telefonech.
- 15.2.6. Aplikace nepodporuje orientaci na šířku (landscape).

## 15.3. Administrace

- 15.3.1. Administrace jsou plně kompatibilní s následujícími prohlížeči v posledních 2 verzích:
  - Google Chrome
  - Mozilla Firefox



- Microsoft Edge
- Safari

15.3.2. Jsou využity technologie standardizované organizacemi jako např. W3C, Ecma International, IEEE atp., které podporují přístupnost a kompatibilitu s různými výstupními zařízeními, tedy typicky validní HTML, CSS, JavaScript atd.

## 16. PRAKTIKY A METODIKY

### 16.1. Automatizace

- 16.1.1. Vývojový proces zahrnuje nástroje a postupy, které zajistí automatizovanou kontrolu dodržování coding standards (linter), případně pre/post procesory , buildovací a balíčkovací nástroje.
- 16.1.2. Všechny konfigurační soubory specifické pro backend (například nastavení webového serveru, nastavení dalších komponent jako třeba Redis, MongoDB, Varnish cache apod.) jsou ukládány a verzovány v Git repozitáři, ke kterému má Zákazník read-only přístup. Tyto soubory se automaticky používají pro konfiguraci serverových součástí; u serverových součástí, kde toto není možné nebo by bylo neadekvátně nákladné, je toto nahrazeno dokumentací k ručnímu nastavení dané součásti.
- 16.1.3. Spolehlivost aplikace a backendu je během vývoje testována minimálně prostřednictvím:
- a. jednotkových testů pro knihovny a modely,
  - b. automatizovaných integračních a API testů.

### 16.2. Internacionalizace a lokalizace

- 16.2.1. JAMB realizován s použitím kódování znaků UTF-8.
- 16.2.2. Všechny texty a zprávy v Aplikaci (např. tlačítka, chybové hlášky) jsou přeložitelné.
- 16.2.3. Aplikace jsou implementovány v české a anglické jazykové verzi.
- 16.2.4. JAMB počítá s budoucím vícejazyčným obsahem (minimálně čeština, angličtina, němčina).
- 16.2.5. Jazyk aplikace se automaticky přizpůsobí jazykovým nastavením zařízení (včetně nastavení pro specifickou aplikaci), pokud je podporován.
- 16.2.6. Aplikace obsahuje jazykový fallback mechanismus – pokud není dostupný překlad v preferovaném jazyce, použije se výchozí jazyk (angličtina). Výjimku tvoří slovenština, pro kterou je výchozí jazyk čeština.
- 16.2.7. Textový obsah aplikace je uložen v samostatných jazykových souborech nebo v databázi (např. JSON, XML, lokalizační formáty jako .strings pro iOS nebo .xml pro Android). Texty nejsou pevně zakódované ve zdrojovém kódu aplikace.
- 16.2.8. Lokalizační klíče jsou konzistentní napříč aplikacemi.

### 16.3. E-maily

- 16.3.1. Všechny e-maily, které jsou posílány jménem (tj. z domén) Zákazníka, jsou zasílány přes Zákazníkem předem schválené SMTP servery či služby, pro které jsou korektně nastavené DNS (SPF, DKIM atp.) záznamy.
- 16.3.2. E-maily odesílané z JAMB nejsou posílány jménem (tj. z domény) u které není zajištěna doručitelnost (SPF, DKIM, DMARC) těchto e-mailů.
- 16.3.3. E-mail obsahuje jméno odesílatele a subject v souladu s aktuálními best practices - např. délka, (ne)použití emojis, interpunkce, verzálek.
- 16.3.4. U HTML e-mailů existuje i TXT verze.
- 16.3.5. U e-mailů se používá preheader ("Johnson Box").
- 16.3.6. E-mail je korektně zobrazen minimálně 90 % příjemcům a v nejpoužívanějších mailových klientech (Gmail, Seznam, Yahoo, Outlook, Apple mail) E-maily nemusí odlišovat "dark-mode" režim pro zobrazení e-mailů.

### 16.4. Další praktiky a metodiky

- 16.4.1. Jsou vybrány a definovány vhodné standardy pro zajištění čistoty zdrojového kódu (coding standards). Popis standardů je součástí dokumentace zdrojového kódu.
- 16.4.2. Zdrojové kódy jsou verzovány pomocí DCVS/ CVS nástroje a uloženy v repozitářích. Zákazník má stálý read-only přístup ke všem těmto repozitářům. Popis verzovacího workflow je součástí dokumentace.
- 16.4.3. Změny zdrojového kódu jsou do repozitářů promítány nejméně 1x týdně.
- 16.4.4. Všechny změny kódu jsou podrobeny Code Review před sloučením do hlavní větve. Review se zaměřuje na kvalitu kódu, bezpečnostní aspekty a dodržování standardů.
- 16.4.5. Každá změna kódu prochází plně automatizovanou pipeline zahrnující:
  - Build
  - Spuštění všech testů.
  - Automatickou kontrolu kvality kódu (linting, statická analýza, kontrola dodržování coding standards).
  - Automatická kontrola zranitelností kódu a závislostí
- 16.4.6. Funkce, které je možné a vhodné (nejen technicky, ale zejména z business logiky) realizovat asynchronně, jsou takto řešeny (např. rozesílání e-mailů).

- 16.4.7. Je použit přístup "Secure by design". Jsou použity frameworky, šablonovací jazyky nebo knihovny, které systémově řeší nedostatky implementace escapováním výstupů a sanitizací vstupů (např. ORM pro přístup k databázi, UI frameworky pro vykreslování DOM).
- 16.4.8. Jsou nasazeny mechanismy monitorování chyb vzniklých za běhu na produkčním prostředí (výjimky, stavy 50x, atp.) a na takto vzniklé chyby Dodavatel automaticky reaguje jako by se jednalo o incident hlášený Zákazníkem.
- 16.4.9. Dodavatel postupuje tak, aby nevznikal zbytečný technický dluh. Technickým dluhem je myšlen zejména:
  - a. důsledek postupů, které v zájmu krátkodobého zvýšení produktivity způsobí vznik provizorních řešení, která přinesou zvýšené náklady na vznik finálního řešení nebo jeho další rozvoj a údržbu;
  - b. důsledek nečinnosti, kdy se zastaráváním použitého řešení zvyšuje nákladnost aktualizace či provozu systému nebo kdy zastaralé řešení bude obsahovat bezpečnostně zranitelné součásti.
- 16.4.10. V případech, kde se vyplatí vědomě technický dluh vytvořit a kde k takovému postupu Zákazník vyjádří souhlas není nutné dodržovat předchozí bod. Dodavatel upozornil Zákazníka na všechny případy, kdy identifikoval, že se vyplatí vytvořit technický dluh.

## 16.5. Zakázané praktiky, technologie a komponenty

- 16.5.1. Zastaralé kryptografické technologie (MD5, SHA-1, SSL, DES, atp.)
- 16.5.2. Zastaralé verze knihoven - jakákoli knihovna, která není udržovaná, nemá aktuální bezpečnostní opravy nebo je naposledy aktualizována před více než třemi roky.
- 16.5.3. Jakýkoliv serverový systém bez aktivní podpory nebo aktualizací.
- 16.5.4. Zastaralé komunikační protokoly (FTP, Telnet, ...).
- 16.5.5. Zastaralé verze programovacích jazyků.
- 16.5.6. Ukládání přihlašovacích údajů v kódu.
- 16.5.7. Nasazování bez CI/CD pipeline nebo jiné automatizace.

## 17. NASAZOVÁNÍ NOVÝCH VERZÍ

### 17.1. Obecné

- 17.1.1. Existuje více prostředí (minimálně vývojové, qa a produkční). Vývojovým prostředím je myšleno typicky lokální vývojové prostředí jednotlivého vývojáře či vnitrofiremní vývojové prostředí dodavatele. QA (Quality Assurance) prostředí je zpřístupněno Zákazníkovi pro testování funkčnosti a jedná se o prostředí technologicky velmi blízké produkčnímu prostředí (s menšími

nároky na výkon a distribuovanost aplikace, pokud toto není předmětem testování). Produkčním prostředím je míněno prostředí veřejně přístupné návštěvníkům a administrátorům webů.

- 17.1.2. Jediné prostředí, které je veřejně přístupné, je produkční prostředí.
- 17.1.3. Součástí procesu vývoje a deploymentu je verzování databázových schémat a nastavení pro migraci dat nebo zajištění stejného či lepšího efektu, který tento požadavek zajišťuje.
- 17.1.4. O všech provedených nasazeních včetně přesných časů je veden záznam v helpdesku a/nebo monitorovacím systému.
- 17.1.5. Postup nasazování na libovolné běhové prostředí je stejný pro všechna prostředí s výjimkou vývojového prostředí a je plně automatizován.
- 17.1.6. Všechny nové verze jsou označeny verzovacími čísly podle standardu Semantic Versioning.
- 17.1.7. Existuje možnost rychlého návratu k předchozí stabilní verzi v případě problému (např. rolling back na backendu nebo deaktivace nové verze aplikace).

## 17.2. Integrační vrstva

- 17.2.1. Nasazování nových verzí integrační vrstvy je prováděno bezvýpadkově (zero-downtime deployment).
- 17.2.2. Po nasazení jsou automaticky spuštěny monitorovací nástroje pro kontrolu klíčových metrik (např. latence API).
- 17.2.3. Systém podporuje postupné nasazování nových verzí (canary deployment nebo blue-green deployment).

## 17.3. Aplikace

- 17.4. Před nasazením nové verze aplikace jsou prováděny automatizované testy na Referenčních zařízeních a relevantních verzích operačního systému (15.2).
- 17.5. Nasazení nové verze aplikace nenarušuje uložená data nebo přihlašovací relace uživatele.
- 17.6. Je možné provádět postupný rollout aplikací.