

Databázové systémy pro chytrou domácnost'

Emma Šlesariková



Univerzita Tomáše Bati ve Zlíně

Fakulta aplikované informatiky

Ústav bezpečnostního inženýrství

Akademický rok: 2024/2025

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení: Emma Šlesariková
Osobní číslo: A22253
Studijní program: B1032A020001 Bezpečnostní technologie, systémy a management
Forma studia: Prezenční
Téma práce: Databázový systém pro chytrou domácnost
Téma práce anglicky: Database System for a Smart Home

Zásady pro vypracování

- Provedte analýzu a vypracujte literární rešerši na dané téma.
- Vypracujte stručný rozbor technologií, které budou v práci použité.
- Provedte analýzu požadavků pro navrhovaný databázový systém.
- Navrhněte databázový systém pro chytrou domácnost na základě požadavků.
- Věnujte pozornost zabezpečení databáze.
- Zhodnotte provedené řešení a navrhněte případné vylepšení.

Forma zpracování bakalářské práce: **tištěná/elektronická**
Jazyk zpracování: **Slovenština**

Seznam doporučené literatury:

1. AUGUSTO, Juan Carlos a NUGENT, Chris D. (ed.). *Designing smart homes: the role of artificial intelligence. Lecture notes in computer science*. Berlin: Springer, c2006. ISBN 9783540359944.
2. DAVIDSON, Louis; MOSS, Jessica. *Pro SQL Server Relational Database Design and Implementation*. Apress, 2016. ISBN 9781484219720.
3. DATE, Chris. *Database Design and Relational Theory: Normal Forms and All That Jazz*. Second edition. Healdsburg, California, U.S.A: APress, 2019. ISBN 978-1-4842-5540-7.
4. LAURENČÍK, Marek. *SQL: podrobný průvodce uživatele*. Praha: Grada Publishing. Průvodce (Grada), 2018. ISBN 978-802-7107-742.
5. POKORNÝ, Jaroslav a Michal VALENTA. *Databázové systémy*. 2. přepracované vydání. Praha: Česká technika – nakladatelství ČVUT, 2020. ISBN 978-80-01-06696-6.

Vedoucí bakalářské práce: **doc. Ing. Zdenka Prokopová, CSc.**
Ústav počítačových a komunikačních systémů
Konzultant bakalářské práce: **Ing. Martin Šrámek**
Ústav bezpečnostního inženýrství

Datum zadání bakalářské práce: **11. prosince 2024**
Termín odevzdání bakalářské práce: **2. června 2025**

doc. Ing. Jiří Vojtěšek, Ph.D. v.r.
děkan



Ing. Jan Valouch, Ph.D. v.r.
ředitel ústavu

Ve Zlíně dne 11. prosince 2024

PROHLÁŠENÍ AUTORA BAKALÁŘSKÉ PRÁCE

Beru na vědomí, že

- odevzdáním bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb., v platném znění bez ohledu na výsledek obhajoby;
- bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému a bude dostupná k nahlédnutí;
- jedno vyhotovení bakalářské práce v listinné podobě bude ponecháno Univerzitě Tomáše Bati ve Zlíně k uložení;
- na moji bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- podle § 60 odst. 1 autorského zákona má Univerzita Tomáše Bati ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- podle § 60 odst. 2 a 3 mohu užít své dílo – bakalářskou práci – nebo poskytnout licenci k jejímu využití jen s předchozím písemným souhlasem Univerzity Tomáše Bati ve Zlíně, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše);
- pokud bylo k vypracování bakalářské práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tj. k nekomerčnímu využití), nelze výsledky bakalářské práce využít ke komerčním účelům;
- pokud je výstupem bakalářské práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá; neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji, že

- jsem na bakalářské práci pracoval(a) samostatně a použitou literaturu jsem řádně citoval(a); v případě publikace výsledků budu uveden(a) jako spoluautor;
- odevzdaná verze bakalářské práce a verze elektronická nahraná do IS/STAG jsou obsahově totožné.

Ve Zlíně, dne

.....

podpis autora

ABSTRAKT

Táto bakalárska práca sa venuje využitiu databázových systémov v chytrých domácnostiach. Skúma, ako sa namerané dáta zo senzorov a ďalších zariadení ukladajú, spracovávajú a využívajú na automatizáciu domácich procesov. Zameriava sa na návrh databáz, používanie jazyka SQL a technológie ako Loxone či Aqara. Práca tiež rieši otázky bezpečnosti a ochrany dát, a zaistenie súkromia užívateľov.

Cieľom je ukázať, ako databázové systémy pomáhajú zvyšovať pohodlie a efektivitu v každodennom živote, a pripraviť základ pre praktickú realizáciu riešenia pre chytrú domácnosť.

Kľúčová slova: Databázové systémy, SQL, inteligentná domácnosť, Loxone

ABSTRACT

This bachelor's thesis focuses on the use of database systems in smart homes. It examines how data from sensors and devices is stored, processed, and utilized to automate household processes. The work emphasizes database design, the use of SQL, and technologies such as Loxone and Aqara. It also addresses issues of data security and protection to ensure user privacy.

The aim is to demonstrate how database systems enhance comfort and efficiency in daily life and to lay the foundation for the practical implementation of a smart home solution.

Keywords: Database system, SQL, smart home, Loxone

OBSAH

ÚVOD.....	13
I TEORETICKÁ ČÁST	14
1 ZÁKLADNÉ PRINCÍPY CHYTREJ DOMÁCNOSTI.....	15
1.1 DEFINÍCIA	15
1.2 TECHNOLOGIE POUŽÍVANÉ V CHYTRÝCH DOMÁCNOSTIACH	15
1.2.1 Riadiace centrum.....	15
1.2.2 Aktéry.....	16
1.2.3 Senzory.....	17
1.2.4 Diaľkové ovládače	19
1.2.5 Hlasový asistenti	19
1.2.6 Prehrávače a ovládače multimédií	20
1.2.7 Sieťové zariadenia.....	20
1.2.8 Nahrávacie zariadenia	21
2 DATABÁZOVÉ SYSTÉMY - ZÁKLADNÉ KONCEPTY.....	22
2.1 VÝZNAM DATABÁZ	22
2.2 POUŽITIE DATABÁZ V RÔZNYCH OBLASTIACH	24
2.3 KLASIFIKÁCIA DATABÁZ – RELAČNÉ A NoSQL	25
2.3.1 Relaçné databázy.....	25
2.3.2 NoSQL databázy	26
2.4 ARCHITEKTÚRA DATABÁZOVÝCH SYSTÉMOV.....	28
3 NÁVRH RELAČNÝCH DATABÁZOVÝCH SYSTÉMOV.....	29
3.1 RELAČNÝ MODEL A JEHO PRINCÍPY	29
3.1.1 Výhody relačného modelu	30
3.1.2 Praktické aplikácie v chytrých domácnostiach	31
4 POUŽITIE SQL V DATABÁZOVÝCH SYSTÉMOCH.....	32
4.1 SQL – DOTAZOVACÍ JAZYK	32
4.2 VYTVÁRANIE TABULIEK	33
5 DATABÁZOVÉ SYSTÉMY V CHYTRÝCH DOMÁCNOSTIACH	35
5.1 ÚLOHA DATABÁZOVÝCH SYSTÉMOV V CHYTRÝCH DOMÁCNOSTIACH	35
5.2 PREPOJENIE DATABÁZOVÝCH SYSTÉMOV S RIADIACIMI JEDNOTKAMI	37
6 BEZPEČNOSŤ A OCHRANA DÁT V DATABÁZOVÝCH SYSTÉMOCH	39
6.1 TECHNIKY ZABEZPEČENIA DÁT.....	39
6.2 HROZBY PRE BEZPEČNOSŤ DATABÁZ	41
6.3 OSVEDČENÉ POSTUPY PRE ZABEZPEČENIE DATABÁZ	42

II	PRAKTICKÁ ČÁST	44
7	NÁVRH RELAČNÝCH DATABÁZ PRE CHYTRÉ DOMÁCNOSTI	45
7.1	NÁVRH DATABÁZOVEJ ŠTRUKTÚRY	45
7.2	TVORBA DIAGRAMU – ERD.....	46
7.3	TVORBA RELAČNEJ SCHÉMY DATABÁZY	48
7.4	TVORBA TABULIEK DATABÁZY	49
7.4.1	Tabuľka Zariadenie	52
7.4.2	Tabuľka Typ_zariadenia.....	53
7.4.3	Tabuľka miestnosť	53
7.4.4	Tabuľka Scéna	54
7.4.5	Tabuľka Zariadenie_Scena.....	54
7.4.6	Tabuľka Akcia.....	55
7.4.7	Tabuľka Typ_akcie	56
7.4.8	Tabuľka Zariadenie_Akcia.....	56
7.4.9	Tabuľka Bezpecnostne_hlasenie	57
7.4.10	Tabuľka Typ_udalosti	58
7.4.11	Tabuľka Zariadenie_Hlasenie	58
7.4.12	Tabuľka Používateľ	59
7.4.13	Tabuľka Rola.....	59
7.4.14	Tabuľka Oprávnenie.....	60
7.4.15	Tabuľka Opravnenie_role.....	60
7.4.16	Tabuľka Senzor	61
7.4.17	Tabuľka Typ_senzora.....	62
7.4.18	Tabuľka Jednotka	62
7.4.19	Tabuľka Senzor_hodnota	63
7.4.20	Tabuľka Zariadenie_senzor	64
8	ZABEZPEČENIE DATABÁZY	65
8.1	POUŽÍVATELSKÉ ÚČTY A PRÍSTUPY V DATABÁZOVOM SYSTÉME.....	65
8.2	ZABEZPEČENIE TABULIEK A DÁT PROSTREDNÍCTVOM OPRÁVNENÍ	66
9	LOXONE CONFIG.....	67
9.1	ROZDELENIE MIESTNOSTI A ICH KONFIGURÁCIA	67
9.1.1	Centrál	67
9.1.2	Kúpeľňa	71
9.1.3	Obývacía izba	73
9.1.4	Spálňa	75
9.1.5	Okolie domu	76
9.2	CVIČENIA PRE ŠTUDENTOV	77
9.2.1	Automatické osvetlenie chodby pomocou pohybového senzora	77
9.2.2	Aktivovať zabezpečovací režim	78
9.2.3	Spustenie alarmu pri narušení priestoru	79
9.2.4	Prístupový systém s rozpoznaním používateľa	80
9.2.5	Zariadenia priradené ku konkrétnej scéne (Príchod domov).....	81
9.2.6	Bezpečnostné hlásenia, ku ktorým sa viažu zariadenia.....	81

9.2.7	Zobrazenie všetkých zariadení v jednotlivých miestnostiach s ich typom a aktuálnym stavom	82
9.2.8	Prehľad zaznamenaných hodnôt senzorov vrátane typu senzora, jednotky času a merania	83
ZÁVER		84
ZOZNAM POUŽITEJ LITERATURY		85
ZOZNAM TABULIEK		87
ZOZNAM OBRÁZKOV		87

ÚVOD

V dnešnej dobe sú chytré domácnosti čoraz populárnejšie, pretože prinášajú do našich životov viac pohodlia, efektívnosti a bezpečnosti. Dôležitú úlohu taktiež zohrávajú aj databázové systémy, ktoré zhromažďujú a spracovávajú dáta z rôznych zariadení, ako sú senzory, osvetlenie, termostaty či bezpečnostné kamery.

Táto práca je písaná práve pre pochopenie toho, ako databázové systémy fungujú v inteligentných domácnostiach. Mojim cieľom nie je len vysvetliť teoretické základy, ale aj ukázať, ako sa tieto technológie dajú využiť v praxi na zlepšenie kvality každodenného života. Či už ide o automatické ovládanie svetiel na základe pohybu, optimalizáciu energetickej spotreby alebo ochranu osobných údajov, databázy zohrávajú kľúčovú úlohu.

V práci sa plánujem venovať viacerým témam – od návrhu a správy databáz, cez ich prepojenie s riadiacimi jednotkami, až po otázky bezpečnosti údajov. Chcem preskúmať, ako tieto systémy dokážu nielen uľahčiť každodenné úlohy, ale aj zvýšiť bezpečnosť a komfort používateľov. Inteligentné domácnosti nie sú len trendom, ale aj ukážkou toho, ako moderné technológie dokážu reálne prispieť k lepšiemu životnému štýlu.

I TEORETICKÁ ČÁST

1 ZÁKLADNÉ PRINCÍPY CHYTREJ DOMÁCNOSTI

Pojem „chytrá domácnosť“ vytvára očakávania prostredia, ktoré je schopné „inteligentne“ reagovať predvídaním, predpovedaním a prijímaním rozhodnutí s prvkami autonómie. Z pohľadu výpočtovej techniky existuje prirodzená asociácia medzi týmito očakávaniami a použitím techník z oblasti umelej inteligencie (AI). Výskum v oblasti inteligentných domov sa v minulosti prevažne sústredil na riešenia orientované na sieťové a hardvérové aspekty. Techniky založené na umelej inteligencii, ktoré podporujú inteligentné správanie, neboli skúmané v rovnakej miere, hoci existujú výnimky. [1]

1.1 Definícia

Definuje sa ako rezidenčné domy so zabudovanou inteligentnou technológiou na zlepšenie úrovne komfortu užívateľov (obyvateľov) zvýšením bezpečnosti a zdravotnej starostlivosti a optimalizáciou spotreby energie. Títo užívatelia môžu ovládať a monitorovať inteligentné domáce spotrebiče na diaľku prostredníctvom systému domáceho manažmentu energie (HEMS), ktorý poskytuje systém vzdialeného monitorovania využívajúci telekomunikačné technológie.

Taktiež sú to akékoľvek rezidenčné budovy, ktoré využívajú rôzne komunikačné schémy a optimalizačné algoritmy na predikciu, analýzu, optimalizáciu a riadenie vzorcov spotreby energie podľa prednastavených preferencií používateľov.

V blízkej budúcnosti budú inteligentné domy zohrávať významnú úlohu nielen ako spotrebitelia energie, ale aj ako dodávatelia energie v moderných sieťach. [2]

1.2 Technológie používané v chytrých domácnostiach

Chytrá domácnosť sa skladá z rôznych komponentov (zariadení). Tieto komponenty umožňujú systémom vykonávať rôzne akcie, sledovať či merať prostredie a tak reagovať na pohyb, otvorenie dverí či okien alebo uskutočniť rôzne udalosti v domácnosti. Každé zariadenie je presne určené na svoju funkciu, ktorú má vykonať v rámci systému a spoločne nám systém dá flexibilitu pri kontrole a ovládaní svojho domova a informuje nás o tom, čo sa deje doma.

1.2.1 Riadiace centrum

Riadiaca jednotka, nazývaná aj ako hub, brána, či gateway, kontroluje a riadi inteligentnú domácnosť. Zvyčajne každá inteligentná domácnosť potrebuje jednu riadiacu

jednotku. Umožňuje pridať a konfigurovať rôzne zariadenia a tiež umožňuje vytvoriť a spustiť “logické udalosti“, nazvané aj ako scény, automatizácie, pravidlá, udalosti alebo nálady.

Centrálna riadiaca jednotka je pripojená k internetu zvyčajne cez Routera pomocou ethernetového kábla alebo WiFi. Pripojenie k internetu nám slúži na to, aby sme mali prístup a mohli ovládať celý systém pomocou mobilného telefónu, tabletu alebo z počítača, a to buď lokálne (v rámci domácej siete) alebo vzdialene (cez Cloudovú službu) odkiaľkoľvek na svete. [3]

Riadiace centrum poskytuje domácim užívateľom vhodné jednotky na monitorovanie s ovládanie rôznych domácich spotrebičov. Všetky dáta v reálnom čase sú zhromažďované systémom na optimalizáciu koordinácie medzi dopytom a výrobou energie.

Hlavné funkcie riadiaceho centra sú tieto nasledovne:

- zhromažďovanie údajov z rôznych meračov, príkazov od domácich užívateľov a z elektrickej siete prostredníctvom vhodného komunikačného systému;
- poskytovanie správneho monitorovania a analýzy spotreby energie v domácnosti pre užívateľov;
- koordinácia medzi rôznymi spotrebičmi a zdrojmi na dosiahnutie optimálneho riešenia preddefinovaných cieľov.

[2]

1.2.2 Aktéry

Aktéry sú zariadenia, ktoré vykonávajú aktívnu činnosť, zmenu, ovládanie či spínanie, pomocou ktorého sa riadi osvetlenie, vykurovanie, elektrické spotrebiče a iné zariadenia. Existujú rôzne druhy aktérov, väčšina z nich sa dajú deliť na:

- Spínače - umožňujú ovládať svetlo alebo elektrické zariadenie spínaním prírodnej elektriky. Typicky spínače majú 2 stavy: zapnuté a vypnuté a sú schopný spínač aj väčšiu záťaž
- Stmievacie - regulujú úroveň svetla alebo výkon regulovateľného zariadenia.
- Ovládače na žalúzie - umožňujú ovládať motorizované žalúzie, rolety, markízy a závesy. Zvyčajne ovládajú stav otvorenia a zatvorenia spínaním správneho signálu.

- Relé moduly - na rozdiel od prvých troch typov zariadení, ktoré zvyčajne sú dotknuteľné pre účel priameho ovládania (vypínač na stene), reléové moduly sú malé moduly, ktoré sa montujú priamo do elektrického vedenia a sú neviditeľné. Zvyčajne umožňujú transformáciu existujúcich prvkov na inteligentné (relé pod vypínač na svetlo).
- Zástrčky - sú najobľúbenejšie prvky, keďže umožňujú rýchlu a jednoduchú transformáciu zariadení na inteligentné. Stačí ich pripojiť do elektrickej zásuvky a do nich pripojiť elektrické zariadenie.
- Zámky na dvere - umožňujú ovládanie a prístup do domu, alebo do určitej časti budovy.
- Smart žiarovky - sem patria LED žiarovky, LED svetlá a LED pásy, ktoré inteligentná domácnosť dokáže priamo ovládať a automatizovať. [3]

1.2.3 Senzory

Senzory sledujú dom, hodnotu, či environmentálny stav a hlásia v prípade, ak detegujú zmenu, ako napríklad pohyb, zmena stavu dverí či okien, alebo špecifické environmentálne ukazovatele ako sú jas, teplota, vlhkosť, CO, dym a podobne.

Typy senzorov:

- Pohybový senzor - informujú systém kedykoľvek zaznamenajú pohyb. Detekcia zvyčajne je zabezpečená pomocou PIR senzoru (Pasívny infračervený senzor), ale v niektorých prípadoch pre vylepšenie detekcie sa používa aj dodatočná mikrovlnná technológia. Zvyčajne sa používajú pri zabezpečujúcich systémoch, alebo na automatizáciu osvetlenia.
- Senzory na dvere či okná - detegujú stav otvorenia a zatvorenia dverí či okien. Zvyčajne sa skladajú z dvoch častí a fungujú na báze magnetickej detekcie. Zvyčajne sa používajú pri zabezpečujúcich systémoch, alebo na automatizáciu osvetlenia.
- Snímače teploty - monitorujú teplotu. Slúžia na automatizáciu na základe nameranej teploty, zvyčajne na reguláciu vykurovania alebo klimatizácie.
- Snímače jasov - tieto senzory slúžia na meranie množstva jasov v miestnosti. Na základe nameranej hodnoty môžu slúžiť v domácej automatizácii ako základ na

určenie, či je deň alebo noc a v sofistikovanejších prípadoch pomáhajú pri automatizácii osvetlenia.

- Multisenzory - sú to také snímače, ktoré v jednom zariadení dokážu snímať viac veličín naraz: pohyb, jas, teplota, stav dverí, vlhkosť, atď. Umožňujú presnejšiu automatizáciu, kde napr. ovládajú svetlo ale iba v prípade, ak v miestnosti je tma.
- Termostaty - sú to kombinované zariadenia a čiastočne patria aj do aktérov. Ako hlavné funkcie majú snímanie teploty a priamu reguláciu vykurovania (majú vstavaný spínač) buď v danej miestnosti, alebo pre dom ako celok v závislosti od rozdielu medzi nameranou a požadovanou hodnotou teploty.
- Záplavové senzory - detegujú vodu alebo inú tekutinu. Hlásia okamžite pri detekcii a tak sú vhodné na skoré zistenie úniku vody. V sofistikovanejšom prípade pomocou domácej automatizácie dokážu priamo regulovať hlavný prívod vody a tak zabrániť vzniku väčšej škody na majetku.
- Detektory dymu - detegujú vzniknutý dym, požiar a v niektorých prípadoch prudký nárast teploty. Umožňujú napríklad spustiť poplach a sirénu. V sofistikovanejších prípadoch dokážu spustiť automatické zhasenie, alebo zapnúť únikové svetlá.
- Detektory CO - detegujú úroveň oxidu uhoľnatého. Oxid uhoľnatý je prudko jedovatý plyn, ktorý vzniká nedokonalým spaľovaním palív na báze uhlíka. Je neviditeľný, nevnímame ho čuchom ani chuťou. Práve preto k každej domácnosti, kde daný typ paliva sa používa na vykurovanie, by malo byť zabezpečený minimálne detektor CO s napojeným na sirénu. [3]
- Detektory CO₂ - detegujú úroveň oxidu uhoľnatého, ktorá v domácnostiach vznikne typicky dýchaním. Každý vieme, že je veľmi dôležité dôkladne vetrať aj pred spaním a aj ráno po zobudení.
- Merače energie - sú to zariadenia, ktoré slúžia na meranie spotreby elektrickej energie buď na určitý spotrebič, vetvu, alebo dokonca na celý dom. Často krát daná funkcia je zabudovaná v rôznych spínačoch a zásuvkách ako dodatočná funkcia. [3]

1.2.4 Diaľkové ovládače

Diaľkové ovládače slúžia ako komfortné prvky, ktoré umožňujú fyzicky ovládať systém bez nutnosti otvárania aplikácie na mobile. Dajú sa programovať, aby ovládali buď konkrétne zariadenie, viac zariadení, alebo aby spustili určitú automatickú scénu.

Diaľkové ovládače delíme na 3 hlavné typy:

- Prenosné – typicky taký diaľkový ovládač je aj ovládač k televízoru. Viete prenášať zo sebou, ale zvyčajne vždy zostane doma.
- Nástenne – zvyčajne sú to batérové zariadenia, ktoré sú upevnené na určitý povrch, či už na stenu, alebo vedľa spotrebiča, ktorú ovládajú.
- Kľúčenky – sú určené na nosenie všade so sebou, sú kompaktné a určené na ovládanie hlavných, „vstupných“ zariadení ako sú brány, alarm alebo vonkajšie osvetlenie. Inteligentná domácnosť má vždy kľúčenku, kde jedna kľúčenka umožňuje ovládať všetky aspekty a vy tak nepotrebuje mať zvlášť kľúčenku na bránu, zvlášť na alarm a na garáž a mať tak veľkú kopu ovládačov, ktoré vám zoberú väčší priestor. [3]

1.2.5 Hlasový asistenti

V posledných rokoch zažili výrazný rozvoj na trhu a dnes sú bežnou súčasťou každého inteligentného zariadenia. Aj keď ich možno ešte nevyužívame, je isté, že v blízkej budúcnosti sa stanú jedným z kľúčových prvkov inteligentných domácností. Teda hlasoví asistenti a inteligentné domácnosti budú neoddeliteľné.

V súčasnosti existujú tri hlavné spôsoby integrácie hlasových asistentov:

1. Integrácia do smartfónov a tabletov – každý moderný smartfón a tablet má zabudovaného hlasového asistenta.
2. Použitie v inteligentných reproduktoroch – tieto zariadenia umožňujú ovládanie inteligentnej domácnosti pomocou hlasu, poskytujú spätnú väzbu alebo odpovedajú na otázky a požiadavky (napríklad informácie o počasi).
3. Podpora v televízoroch – niektoré moderné televízory, ako napríklad Android TV, obsahujú podporu hlasových asistentov. [3]

1.2.6 Prehrávače a ovládače multimédií

V súčasnosti čoraz viac ľudí vyžaduje integráciu multimediálnych a audiovizuálnych zariadení do svojich inteligentných domácností. Cieľom je mať možnosť jednoducho spravovať hudbu aj video obsah. Napríklad, jednoduchým automatizačným pravidlom „filmový režim“ by sa mohlo nielen upraviť osvetlenie a polohu žalúzií, ale zároveň aj zapnúť televízor, spustiť obľúbenú službu ako YouTube či Netflix a aktivovať domáci audio systém.

V súčasnosti sa multimediálne systémy dajú ovládať dvoma hlavnými spôsobmi:

- Infračervené ovládače – využívajú technológiu infračerveného prenosu na ovládanie zariadení.
- Sieťové ovládanie – zariadenia sú pripojené k domácej sieti, čo umožňuje ich ovládanie cez aplikácie alebo hlasových asistentov.

[3]

1.2.7 Sieťové zariadenia

Najčastejšie sa používa sieťové zariadenie v podobe IP kamier a sieťových nahrávačov videa. IP kamery sa stávajú čoraz populárnejšími a to preto, že výrazne prispievajú k zvýšeniu úrovne bezpečnosti a poskytujú pocit kontroly nad domácnosťou.

1. **Vonkajšie IP kamery:** Sú navrhnuté tak, aby odolávali vonkajším podmienkam. Buď sú vybavené odolnými krytmi, alebo je ich inštalácia navrhnutá pod strechou či inými ochrannými prvkami. Súčasne IP kamery dosahujú vysokú úroveň kvality, pričom už existujú modely s rozlíšením 4K a dokonca aj také, ktoré dokážu v noci zaznamenať farebný obraz namiesto čiernobieleho. Niektoré kamery sú vybavené vstavanou Wi-Fi, kdeže nie vždy je možné použiť sieťový kábel. Ale ideálnym riešením je použitie technológie POE (Power over ethernet), ktorá umožňuje prostredníctvom jedného sieťového kábla zabezpečiť napájanie kamery a zároveň aj jej pripojenie do siete.
2. **Vnútorne kamery:** Sú konštruované tak, aby mohli byť umiestnené vo vnútri. To znamená, že nemajú takú kvalitnú ochranu voči počasiu, ale to ani nie je potrebné. Rovnako ako vonkajšie kamery, aj tie vnútorné môžu podporovať aj WiFi sieť, ale stále platí pravidlo, čím kvalitnejšia kamera, tým menej pravdepodobné, že WiFi bude dostačujúce na kvalitné video. [5]

1.2.8 Nahrávacie zariadenia

Tieto zariadenia slúžia na nahrávanie záznamov z IP kamier a líšia sa hlavne v počte kamier, ktoré dokážu naraz spracovať. Bežné modely umožňujú pripojenie 4 alebo 8 kamier, čo je pre väčšinu domácností postačujúce. Na nahrávanie je potrebný harddisk, ktorý sa dokupuje samostatne. Platí jednoduché pravidlo: čím väčší harddisk, tým dlhší časový úsek záznamu, ktorý môžete uchovať.

Ďalším dôležitým faktorom je, či má nahrávač zabudované POE (Power over Ethernet) výstupy pre kamery. Táto funkcia môže zvýšiť cenu nahrávača, ale eliminuje potrebu dodatočného POE switcha, čo je praktické najmä v priestorovo obmedzených podmienkach alebo pri nedostatku sieťových pripojení. [3]

2 DATABÁZOVÉ SYSTÉMY - ZÁKLADNÉ KONCEPTY

Databáza je usporiadaná zbierka štruktúrovaných informácií alebo údajov, ktoré sú spravidla uložené elektronicky v počítačovom systéme. Riadi ich systém správy databáz, známy ako DBSM (Database management systém). Kombinácia údajov, DBSM a aplikácií, ktoré s nimi spolupracujú, sa nazýva databázový systém, často skráteno označovaný ako databáza.

Najbežnejšie typy databáz dnes ukladajú údaje vo forme riadkov a stĺpcov usporiadaných do tabuliek. Tento model umožňuje jednoduché spracovanie a efektívne vyhľadávanie údajov. Takto uložené údaje je možno ľahko vyhľadávať, spravovať, upravovať a aktualizovať. Väčšina databáz používa pre tieto účely jazyk SQL (Structured Query Language). [4]

2.1 Význam databáz

1. Zlepšená bezpečnosť údajov

Bezpečnosť je kľúčová pre každú organizáciu, ktorá pracuje s dôvernými údajmi. Systém správy databáz (DBMS) poskytuje silné bezpečnostné prvky, ako sú šifrovanie, autentifikácia používateľov a kontrola prístupu. Tieto funkcie umožňujú administrátorom určiť, kto môže údaje mazať, pristupovať k nim alebo ich upravovať. DBMS zároveň podporuje protokolovanie a auditovanie, čo umožňuje monitorovať aktivity používateľov, detegovať bezpečnostné incidenty a zabezpečiť údaje v súlade s právnymi normami. [5]

2. Efektívny prístup k údajom

DBMS umožňuje organizované získavanie údajov prostredníctvom SQL. Používatelia môžu vytvárať zložité dotazy na triedenie, filtrovanie a analýzu dát. Optimalizácia dotazov minimalizuje čas odozvy a zjednodušuje prácu organizáciám. Mechanizmy ako cacheovanie a indexovanie urýchľujú prístup k údajom, čo zabezpečuje, že používatelia majú informácie dostupné bez zbytočných oneskorení. [5]

3. Integrita údajov

DBMS zaisťuje, že údaje zostávajú konzistentné a presné. Uplatňuje pravidlá, ako sú primárne a cudzie kľúče, čím predchádza nezrovnalostiam, ako sú neplatné záznamy alebo duplicity. Táto konzistencia je zásadná, najmä ak k údajom pristupuje viac používateľov súčasne. Systém správy databáz tiež dohliada na správne vykonávanie transakcií bez chýb. [5]

4. Škálovateľnosť

S rastom organizácie je potrebné zvládnuť rastúci objem údajov. DBMS je navrhnutý tak, aby efektívne zvládol zvýšenie počtu údajov a používateľov bez straty výkonu. Podporuje vertikálne aj horizontálne škálovanie, čo umožňuje zvládnuť veľké zaťaženie a zložité dotazy. Tento aspekt robí DBMS vhodným pre organizácie rôznych veľkostí – od malých po veľké podniky. [5]

5. Centralizovaná správa údajov

DBMS eliminuje roztrieštené zdroje údajov tým, že všetky dáta sú uložené na jednom mieste. To uľahčuje správu údajov a zabezpečuje, že všetci používatelia pracujú s rovnakými aktuálnymi informáciami. Centralizácia znižuje náklady na úložisko, eliminuje redundantné údaje a podporuje správnu správu dát. [5]

6. Obnova a zálohovanie údajov

Strata alebo nepresnosť údajov môže mať vážne dôsledky. DBMS poskytuje zabudované mechanizmy na zálohovanie a obnovu údajov, ktoré zabezpečujú pravidelné zálohovanie a obnovu v prípade nepredvídaných udalostí. Funkcie ako protokolovanie transakcií a automatické plánovanie záloh umožňujú organizáciám obnoviť údaje do presného okamihu, čím sa minimalizuje strata a prestoje. [5]

7. Optimalizované zdieľanie údajov

DBMS umožňuje jednoduché zdieľanie údajov medzi rôznymi oddeleniami organizácie. Podporuje viacpoužívateľské prostredia, kde viacerí používatelia môžu súčasne pracovať s rovnakými údajmi. Táto schopnosť zlepšuje spoluprácu a umožňuje tímom prístup k potrebným informáciám na jednom mieste. [5]

8. Nezávislosť údajov

Jednou z hlavných výhod DBMS je nezávislosť údajov. To znamená, že štruktúru databázy je možné meniť bez ovplyvnenia aplikácií, ktoré s ňou pracujú. Vývojári môžu napríklad upraviť typy údajov bez potreby prepisovať kód aplikácie, čo šetrí čas a náklady na údržbu a aktualizácie. [5]

9. Vyššia rýchlosť a efektivita

DBMS automatizuje mnoho úloh spojených so správou údajov, ako je optimalizácia dotazov, indexovanie a správa úložiska. Tieto procesy zrýchľujú spracovanie a získavanie údajov, čím sa zvyšuje celková efektivita. [5]

10. Regulačná zhoda

Dodržiavanie regulačných požiadaviek je pre firmy spracúvajúce citlivé údaje nevyhnutné. DBMS pomáha zabezpečiť súlad s predpismi poskytovaním auditných záznamov, sledovaním zmien v údajoch a dodržiavaním pravidiel uchovávanía údajov. Tým sa znižuje riziko právnych sankcií a zvyšuje dôveryhodnosť organizácie. DBMS teda slúži ako základ pre moderné technológie, pričom poskytuje konzistenciu, bezpečnosť a integritu údajov. To z neho robí nevyhnutný nástroj na správu údajov v organizáciách. [5]

2.2 Použitie databáz v rôznych oblastiach

Databázy sa využívajú v mnohých oblastiach, kde zohrávajú kľúčovú úlohu pri správe a analýze údajov. V bankovníctve a finančných službách uchovávajú databázy údaje o klientoch, účtoch a transakciách, čo umožňuje rýchly prístup k finančným operáciám, zaručenú presnosť a bezpečnosť. Pomáhajú tiež pri detekcii podvodov analýzou údajov v reálnom čase na identifikáciu podozrivých aktivít.

V oblasti e-commerce databázy slúžia na uchovávanie informácií o produktoch, ich dostupnosti a cenách, spracovanie objednávok a transakcií. Personalizácia odporúčaní pre zákazníkov je možná vďaka analýze ich nákupného správania, čo zlepšuje celkovú zákaznícku skúsenosť.

Zdravotníctvo ich využíva na správu elektronických zdravotných záznamov, monitorovanie pacientov prostredníctvom senzorov a podporu výskumu a analýzy medicínskych dát. Tieto funkcie prispievajú k lepšej zdravotnej starostlivosti a efektívnejšiemu využívaniu údajov.

V sektore vzdelávania nachádzajú databázy uplatnenie pri správe údajov o študentoch, kurzoch, známkach či dochádzke. Knižničné systémy využívajú databázy na evidenciu kníh a ich dostupnosti, zatiaľ čo e-learningové platformy ich používajú na spravovanie online kurzov. [6]

Telekomunikácie spravujú databázy na evidenciu uskutočnených hovorov a správ, riadenie zákazníckych profilov a optimalizáciu sietí. Analýza dát zlepšuje výkon telekomunikačných systémov a zvyšuje spokojnosť zákazníkov.

Výroba a priemysel si databázy cenia pri riadení zásob, optimalizácii výrobných procesov a sledovaní dodávateľských reťazcov. Efektívna logistika a plánovanie výroby sú vďaka týmto systémom jednoduchšie.

V médiách a zábave slúžia databázy na uchovávanie obsahu pre streamovacie služby, ako sú filmy a hudba. Personalizácia obsahu na základe správania používateľov a cielené reklamné systémy umožňujú poskytovať lepšie služby.

Chytré domácnosti a IoT spravujú údaje zo senzorov, ktoré sledujú teplotu, svetlo či pohyb, a umožňujú automatizáciu procesov, ako je riadenie osvetlenia a kúrenia. Optimalizácia energetickej spotreby je ďalším prínosom.

Vo verejnej správe a samospráve nachádzajú databázy využitie pri evidencii obyvateľov, daňovníkov a voličov. Slúžia aj na zhromažďovanie údajov pre dopravné systémy či štatistické analýzy, ktoré podporujú plánovanie a rozhodovanie.

Oblasť obrany a bezpečnosti sa spolieha na databázy pri riadení údajov o vojakoch, vybavení a operáciách, monitorovaní hrozieb a zabezpečení citlivých údajov prostredníctvom šifrovania. Tieto funkcie zaisťujú ochranu a bezpečnosť údajov. [6]

2.3 Klasifikácia databáz – relačné a NoSQL

2.3.1 Relačné databázy

„Definíciou relačnej databázy môžeme rozumieť množinu deklarácií tabuliek, pohľadov, integrovaných obmedzení, domén a prípadne ďalších prvkov schémy. Obecne môže byť relačná databáza tvorená niekoľkými schémami.“ [7]

Predstavujú jeden z najrozšírenejších modelov na ukladanie a správu dát. Sú založené na prístupe, ktorý dáta organizuje vo forme tabuliek (relačných schém), pričom každá tabuľka obsahuje údaje štruktúrované do riadkov a stĺpcov.

Základné prvky relačnej databázy:

1. Tabuľka:

- Tabuľky predstavujú základnú štruktúru na ukladanie údajov, pričom každý riadok reprezentuje jeden záznam (entitu) a každý stĺpec reprezentuje jeho vlastnosti (atribúty). Jednotlivé tabuľky sú prepojené pomocou vzťahov, čo umožňuje efektívne organizovanie a prepojenie veľkého množstva údajov.
2. Integrita a referenčné väzby:
- Relačné databázy implementujú mechanizmy na zabezpečenie integrity dát, napríklad referenčné kľúče, ktoré zaručujú konzistentnosť medzi tabuľkami
 - Napríklad ak existuje tabuľka „filmy“ a tabuľka „žánre“, vzťah medzi nimi zaisťuje, že každý záznam v tabuľke „filmy“ odkazuje na platný záznam v tabuľke „žánre“.
3. Viacnásobné schémy:
- Databáza môže obsahovať viacero relačných schém, ktoré reprezentujú rozdielne aspekty spracovaných dát.

Relačné databázy sú obzvlášť užitočné pre organizácie, ktoré potrebujú ukladať a analyzovať veľké množstvo prepojených dát. Ich štruktúrovaný model umožňuje jednoducho vizualizovať dáta a to hlavne pre lepšie pochopenie dátovej štruktúry, kde môžu vývojári používať vizualizačné nástroje, ktoré uľahčujú návrh a správu databáz. Štruktúrovaný model nám taktiež umožňuje modifikáciu existujúcich schém, v prípade potreby je možné upraviť databázové schémy tak, aby lepšie reflektovali meniace sa požiadavky systému. [8]

2.3.2 NoSQL databázy

„Termín NoSql databáza bola zvolená pre voľne špecifikovanú triedu nerelačných dátových úložísk. Takéto databázy nepoužívajú SQL ako svoj dotazovací jazyk. Termín NoSQL je preto zavádzajúci a v databázovej komunite je vykladaný skôr ako „not only SQL“.“ [7]

„Časť NoSQL databáze obvykle zjednodušuje alebo obmedzuje réžiu vyskytujúcu sa v relačných databázach s plnou funkčnosťou. Ich dáta sú často organizované na logickej úrovni do istých aj viacrozmerých tabuliek a prístupované len prostredníctvom primárneho kľúča.“ [7]

Bez ohľadu na rôzne obmedzenia umožňujú NoSQL databázy vyvíjať užitočné aplikácie.

Jednoduchšie povedané NoSQL databázy predstavujú moderný prístup k spracovaniu a ukladaniu dát, ktorý sa líši od tradičných relačných databáz. Sú navrhnuté na prácu s veľkým objemom neštruktúrovaných alebo polo štruktúrovaných dát a poskytujú flexibilitu, ktorú tradičné modely často nedokážu zabezpečiť. [7]

Dátový model NoSQL databázy

Na rozdiel od klasických databáz, ktoré sa spoliehajú na striktný logický dátový model, sú NoSQL databázy oveľa flexibilnejšie. Používajú intuitívny prístup k organizácii dát, pričom pracujú s nasledujúcimi konceptmi:

- **Páry kľúč – hodnota:** najjednoduchší typ dátového modelu, kde je hodnota uložená ako dvojica (kľúč, hodnota). Tento prístup umožňuje jednoduché a rýchle vyhľadávanie údajov na základe kľúča.
- **Dokumentované databázy:** využívajú formát JSON alebo XML na ukladanie komplexných štruktúr dát, pričom každá položka môže obsahovať hierarchické informácie.
- **Grafové databázy:** umožňujú prácu s údajmi, ktoré majú vzťahy medzi sebou. Tieto databázy sú vhodné na reprezentáciu sociálnych sietí alebo navigáciu.

Výhody NoSQL databáz:

1. **Flexibilita:** Schéma dát nie je striktno definovaná, čo umožňuje jednoduchšie zmeny a rozšírenia.
2. **Rýchlosť:** Dotazy v NoSQL databázach bývajú spracované efektívne, najmä pri veľkom množstve neštruktúrovaných dát.
3. **Škálovateľnosť:** Možnosť horizontálneho rozšírenia (pridávanie ďalších serverov) umožňuje prácu s obrovskými objemami dát. [7]

NoSQL databázy predstavujú moderné riešenie pre aplikácie, ktoré vyžadujú vysokú dostupnosť, flexibilitu a rýchlosť. Ich použitie je čoraz častejšie a to najmä v oblasti IoT a cloudových služieb.

Dotazovanie v NoSQL databázach je ich najmenej prepracovanou zložkou. Líši sa to od tradičného SQL štandardu a často využíva vlastné API alebo rozšírenia SQL. API nám znižuje záťaž na server a umožňuje efektívnu distribúciu úloh medzi klientom a serverom.

Pri práci s databázami ako CouchDB alebo Google AppEngine sa využívajú špecifické dotazovacie jazyky. Jedným z nich je GQL (Google Query Language), ktorý je podobný SQL a využíva sa na prácu s dokumentovými databázami. Ďalším dotazovacím jazykom je API volanie, ktoré vykonávajú priame volanie pre CRUD operácie (vytváranie, čítanie, aktualizácia a mazanie dát). Tento štýl dotazovania je vhodný pre jednoduché úlohy, ako je vyhľadávanie kľúčových slov alebo filtrovanie podľa parametrov. Naopak, môže byť časovo náročný pri komplexných operáciách, ktoré vyžadujú pokročilé spracovanie dát. [7]

2.4 Architektúra databázových systémov

Architektúra databázových systémov predstavuje špecializovanú oblasť informatiky, ktorá sa zameriava na návrh štruktúry databázového systému a jeho komponentov. Architektúra taktiež zahŕňa konkrétnu štruktúru (schému) databázového systému, ktorá zobrazuje jeho základnú organizáciu. Tento koncept ponúka abstraktné metódy popisu štruktúry databázových systémov a poskytuje princípy a pravidlá pre ich návrh a vývoj.

Jednou z najznámejších foriem je trojúrovňová architektúra, známa aj ako ANSI/SPARC architektúra. Táto architektúra bola navrhnutá na oddelenie programov od údajov, podporu viacerých používateľských pohľadov a využitie údajového slovníka (katalógu), ktorý uchováva opis databázovej štruktúry. [9]

ANSI/SPARC architektúra definuje tri úrovne schém:

1. **Vnútoraná úroveň:** obsahuje vnútornú schému, ktorá popisuje fyzické štruktúry pamäte v databáze. Používa fyzický údajový model na opis detailov ukladania údajov a prístupu k nim.
2. **Konceptuálna úroveň:** Zahŕňa konceptuálnu schému, ktorá poskytuje globálny pohľad na štruktúru celej databázy. Táto schéma skrýva detaily fyzického uloženia údajov a zameriava sa na opis entít, typov údajov, vzťahov a obmedzení. Na jej vytvorenie sa používa vysokoúrovňový alebo implementačný údajový model.
3. **Externá (vonkajšia) úroveň:** Zameriava sa na používateľské pohľady (externé schémy), pričom každý pohľad popisuje konkrétnu časť databázy, ktorá je relevantná pre určitú skupinu používateľov.

Táto architektúra umožňuje flexibilnejšie spravovanie údajov, zjednodušuje prácu s databázou a zvyšuje bezpečnosť tým, že používatelia vidia len tie dáta, ktoré sú pre nich relevantné. [9]

3 NÁVRH RELAČNÝCH DATABÁZOVÝCH SYSTÉMŮ

3.1 Relačný model a jeho princípy

Relačný model bol navrhnutý Edgarom F. Coddom v 70. rokoch 20. storočia ako teoretický základ pre organizáciu a manipuláciu s dátami v databázových systémoch. Tento model je známy svojou precíznou definíciou dát a ich štruktúry, ktorá poskytuje formálny základ pre návrh, správu a manipuláciu s databázami. Podľa Chrisa Datea, "relačný model predstavuje konzistentný a robustný spôsob reprezentácie dát v digitálnych systémoch, ktorý zaručuje logickú konzistenciu a zjednodušuje prácu s dátami" [10]

Základné princípy relačného modelu:

1. Dáta ako vzťahy

- Relačný model reprezentuje dáta vo forme tabuliek, ktoré pozostávajú z riadkov (záznamov) a stĺpcov (atribútov). Každá tabuľka reprezentuje vzťah medzi entitami. Date uvádza, že "relácie, ako sú tabuľky, sú logické a umožňujú jasne definovať vzťahy medzi entitami bez redundancie" [10]
- Riadky: Reprezentujú jednotlivé záznamy (napr. konkrétne zariadenie v chytnej domácnosti).
- Stĺpce: Definujú vlastnosti entít (napr. názov zariadenia, jeho aktuálny stav, čas poslednej aktualizácie).

2. Jednoznačné identifikátory (kľúče)

- Každý riadok v tabuľke je jednoznačne identifikovaný pomocou primárneho kľúča, ktorý zabezpečuje integritu dát. Ako Date zdôrazňuje, "primárne kľúče sú nevyhnutné na zaistenie toho, že každý riadok je unikátny a jasne rozlíšiteľný" [10]

3. Integrácia dát

Relačný model poskytuje dva hlavné mechanizmy integrity:

- Entity Integrity: Každý záznam musí mať jedinečný primárny kľúč, ktorý zaručuje, že údaje sú konzistentné.
- Referenčná integrita: Podľa Datea, "referenčná integrita zaručuje, že všetky cudzí kľúče odkazujú na platné záznamy, čím sa eliminuje riziko nekonzistentných dát medzi tabuľkami" [10]

4. Normalizácia

- Normalizácia je proces rozkladu tabuliek na menšie štruktúry s cieľom eliminovať redundanciu a zlepšiť konzistenciu dát. Date opisuje normalizáciu ako "základný krok pri navrhovaní relačných databáz, ktorý minimalizuje duplicitu dát a optimalizuje výkon" [10]. Normalizácia prebieha cez rôzne úrovne, napríklad prvá, druhá a tretia normálna forma.

5. Manipulácia s dátami pomocou SQL

- Relačný model podporuje používanie štandardizovaného jazyka SQL na interakciu s dátami. Ukážka príkazov jazyka SQL:
 - INSERT: Vkladanie nových záznamov.
 - SELECT: Načítanie existujúcich dát.
 - UPDATE: Aktualizácia existujúcich údajov.
 - DELETE: Mazanie nepotrebných údajov.

3.1.1 Výhody relačného modelu

Relačný model prináša množstvo výhod, ktoré ho robia ideálnym riešením pre organizáciu a manipuláciu s dátami v rôznych oblastiach. Jednou z hlavných výhod je jeho jednoduchosť – dáta sú usporiadané vo forme tabuliek, čo umožňuje intuitívne pochopenie štruktúry a jednoduchú správu údajov. Tabuľky a ich prepojenia jasne znázorňujú vzťahy medzi rôznymi entitami, čo uľahčuje ich vizualizáciu a analýzu.

Ďalšou výhodou je flexibilita. Relácie medzi tabuľkami môžu byť ľahko upravované, rozširované alebo kombinované bez toho, aby sa narušila celková štruktúra databázy. Tento prístup umožňuje databázovým správcom rýchlo reagovať na meniace sa požiadavky používateľov alebo systémov.

Mechanizmy integrity, ktoré relačný model poskytuje, zaisťujú konzistenciu dát. Entity integrity a referenčná integrita chránia pred nekonzistenciou a zabezpečujú, že všetky uložené údaje sú správne a spoľahlivé. Navyše, relačné databázy ponúkajú škálovateľnosť, ktorá umožňuje prispôsobenie systémov tak, aby zvládali malé aj veľké objemy dát. [10]

3.1.2 Praktické aplikácie v chytrých domácnostiach

Relačné databázy nachádzajú významné uplatnenie aj v oblasti inteligentných domácností, kde je potrebné spracovať veľké množstvo dát generovaných rôznymi senzormi a zariadeniami. Tabuľky umožňujú efektívnu organizáciu týchto údajov, pričom vzťahy medzi tabuľkami zabezpečujú ich logickú prepojenosť. Napríklad tabuľka „Senzory“ môže obsahovať informácie o jednotlivých senzoroach, ako sú teplota, pohyb alebo vlhkosť, zatiaľ čo tabuľka „Zariadenia“ môže ukladať údaje o konkrétnych zariadeniach, ako sú termostaty alebo osvetlenie.

Tieto dáta môžu byť prostredníctvom cudzích kľúčov prepojené, čo umožňuje vytvárať sofistikované automatizačné pravidlá. Napríklad informácia zo senzora pohybu v miestnosti môže byť spojená s údajmi o osvetlení, čo umožní automatické zapnutie svetiel pri detekcii prítomnosti osoby. Podobne môžu byť údaje o teplote prepojené s termostatom, ktorý prispôsobí vykurovanie podľa aktuálnych podmienok.

Relačné databázy teda umožňujú nielen efektívnu organizáciu dát, ale aj ich analýzu a prepojenie, čím prispievajú k vyššiemu komfortu, energetickej efektivite a automatizácii v chytrých domácnostiach. Tento model poskytuje robustnú platformu pre inteligentné systémy, ktoré využívajú prepojené zariadenia na zlepšenie kvality života ich používateľov.

[11]

4 POUŽITIE SQL V DATABÁZOVÝCH SYSTEMOCH

4.1 SQL – dotazovací jazyk

SQL (Structured Query Language) je štandardizovaný jazyk určený na prácu s dátami v relačných databázových systémoch. Umožňuje efektívne vytvárať, upravovať a spravovať štruktúru databáz, ako aj manipulovať s údajmi v nich uloženými. Podľa Mareka Laurenčíka je SQL „štandardizovaný a robustný nástroj, ktorý umožňuje manipulovať s dátami a spravovať štruktúru databáz tak, aby reflektovala potreby moderných aplikácií“. [12]

Základné funkcie SQL

SQL sa skladá z niekoľkých podjazykov, ktoré sa zameriavajú na rôzne aspekty práce s databázami:

1. **DDL (Data Definition Language):** Podľa Laurenčíka „DDL definuje štruktúru databázy, vrátane tabuliek, indexov a ďalších objektov, ktoré tvoria jej základ“. [12]

Príklady:

- CREATE TABLE: Na vytvorenie novej tabuľky.
 - ALTER TABLE: Na úpravu štruktúry existujúcej tabuľky.
 - DROP TABLE: Na odstránenie tabuľky.
2. **DML (Data Manipulation Language):** Tieto príkazy slúžia na manipuláciu s údajmi uloženými v tabuľkách. Laurenčík uvádza, že „DML umožňuje pracovať s dátami prostredníctvom príkazov, ako je INSERT, UPDATE a DELETE“. [12]
 3. **DQL (Data Query Language):** Používa sa na získavanie a filtrovanie údajov z databáz. „Príkaz SELECT je základným nástrojom pre čítanie dát a ich spracovanie na základe špecifických kritérií“. [12]
 4. **DCL (Data Control Language):** Tento jazyk slúži na správu prístupových práv v databáze. „Príkazy GRANT a REVOKE umožňujú kontrolovať, kto má prístup k určitým údajom alebo objektom v databáze“. [12]

Význam SQL v inteligentných domácnostiach

Použitie SQL v chytrých domácnostiach umožňuje:

1. **Efektívnu správu údajov:** SQL ponúka nástroje na ukladanie, načítavanie a aktualizáciu dát zo senzorov a zariadení.
2. **Škálovateľnosť:** „SQL je schopný spracovať veľké množstvo dát v reálnom čase, čo ho robí ideálnym riešením pre aplikácie v oblasti IoT“. [12]
3. **Kompatibilitu:** SQL je štandardizovaný jazyk, ktorý podporuje väčšina databázových systémov, čo uľahčuje jeho implementáciu v rôznych projektoch.

4.2 Vytváranie tabuliek

Vytváranie a správa tabuliek v SQL je základnou funkciou databázy. Tabuľky slúžia na organizáciu dát do riadkov a stĺpcov, čo umožňuje efektívne ukladanie a manipuláciu s informáciami.

Na vytvorenie novej tabuľky sa používa príkaz CREATE TABLE. Pri definovaní tabuľky je potrebné špecifikovať názvy stĺpcov a ich dátové typy. Napríklad, ak chceme vytvoriť tabuľku pre ukladanie informácií o používateľoch, môžeme použiť nasledujúci príkaz:

```
CREATE TABLE Pouzivatelias (  
    Id INT NOT NULL PRIMARY KEY IDENTITY,  
    Meno NCHAR(60) NULL,  
    Priezvisko NCHAR(60) NULL,  
    Datum_narodenia DATE NULL,  
    Pocet_clankov INT NULL  
);
```

V tomto príkaze definujeme tabuľku s názvom Pouzivatelias, ktorá obsahuje stĺpce Id, Meno, Priezvisko, Datum_narodenia a Pocet_clankov. Stĺpec Id je definovaný ako primárny kľúč s automatickým inkrementovaním hodnoty.[8]

Dôležité aspekty pri vytváraní a správe tabuliek:

- **Názvy stĺpcov a tabuliek:** Odporúča sa používať konzistentné a zrozumiteľné názvy bez diakritiky, pričom slová v názvoch môžu byť oddelené podčiarkovníkom (_).

- **Dátové typy:** Správny výber dátových typov pre jednotlivé stĺpce je kľúčový pre optimalizáciu výkonu a efektívne využitie úložného priestoru.
- **Obmedzenia (constraints):** Definovanie obmedzení, ako sú primárne kľúče, unikátne kľúče a not null, zabezpečuje integritu dát v tabuľkách.

[8]

5 DATABAZOVÉ SYSTÉMY V CHYTRÝCH DOMÁCNOSTIACH

5.1 Úloha databázových systémov v chytrých domácnostiach

Databázové systémy predstavujú kľúčový komponent inteligentných domácností, pretože umožňujú efektívne spracovanie, organizáciu a správu dát generovaných rôznymi zariadeniami a senzormi. Augusto a Nugent definujú tieto systémy ako „infraštruktúra, ktorá integruje dáta z rôznych zdrojov, aby podporovala automatizáciu a zlepšila komfort užívateľov“. [1]

Základné funkcie databázových systémov v chytrých domácnostiach:

1. Zbieraní dát zo senzorov a zariadení:

- Inteligentné domácnosti generujú obrovské množstvo dát, napríklad informácie o teplote, vlhkosti, spotrebe energie alebo pohybe. Tieto dáta sú zbierané senzormi a ukladané do centrálnej databázy, kde sú dostupné pre ďalšie spracovanie. Napríklad teplotný senzor môže poskytnúť údaje, ktoré sú využívané na riadenie vykurovacieho systému.

2. Správa uložených dát:

- Podľa Augusto a Nugenta „presná a spoľahlivá správa dát je nevyhnutná pre efektívnu automatizáciu“. [1] Databázové systémy poskytujú mechanizmy na aktualizáciu a mazanie zastaraných dát, čo zaručuje konzistenciu a presnosť informácií.

3. Analýza a interpretácia dát:

- Moderné databázové systémy dokážu vykonávať analýzu dát v reálnom čase. Napríklad môžeme identifikovať vzory v spotrebe energie, ktoré môžu byť použité na optimalizáciu prevádzky inteligentnej domácnosti.

Databázové systémy slúžia ako integračný bod pre rôzne zariadenia a platformy. Riadiace jednotky, ako je Loxone Miniserver, zhromažďujú dáta z pripojených zariadení a ukladajú ich do centrálnej databázy. Tieto systémy umožňujú:

- **Automatizované rozhodovanie:** Na základe zhromaždených dát môže systém samostatne vykonávať akcie, napríklad zapínať osvetlenie, keď detekuje pohyb.
- **Vzdialený prístup:** Používatelia môžu kontrolovať stav domácnosti a meniť nastavenia cez mobilné aplikácie.
- **Kompatibilitu:** Integrácia s technológiami tretích strán, ako je Zigbee alebo Z-Wave, rozširuje možnosti systému. [13]

Praktické aplikácie databáz v chytrých domácnostiach

1. Monitorovanie spotreby energie:

- Databázový systém uchováva záznamy o spotrebe energie v jednotlivých miestnostiach alebo zariadeniach. Tieto informácie sú používané na identifikáciu najväčších spotrebiteľov a optimalizáciu ich prevádzky.

2. Zabezpečenie:

- Informácie zo senzorov, ako sú kamery alebo pohybové detektory, sú ukladané a analyzované na ochranu domácnosti pred narušením.

3. Riadenie osvetlenia:

- Na základe uložených dát môže systém upravovať intenzitu svetla v miestnostiach podľa prírodného osvetlenia alebo prítomnosti osôb.

Podľa Augusta a Nugenta „budúcnosť databázových systémov spočíva v ich schopnosti integrovať sa s umelou inteligenciou a poskytovať prediktívne analýzy“. [1] Táto integrácia umožní inteligentným domácnostiam nielen reagovať na aktuálne podmienky, ale aj predvídať budúce potreby svojich používateľov. Týmto spôsobom sa zvýši ich schopnosť prispôbiť sa individuálnym požiadavkám, čo prispeje k ešte efektívnejšiemu využívaniu technológií v každodennom živote.

5.2 Prepojenie databázových systémov s riadiacimi jednotkami

Inteligentné domáce systémy, ako sú Loxone a Aqara, predstavujú moderné riešenia, ktoré efektívne prepájajú databázové systémy s riadiacimi jednotkami na správu údajov z rôznych senzorov, detektorov alebo iných zariadení. Tieto systémy umožňujú zbierať dáta, ukladať ich do databáz a využívať ich na analýzu a automatizáciu procesov, čím zvyšujú komfort, bezpečnosť a efektivitu domácností.

Ako uvádzajú Augusto a Nugent: „Budúcnosť databázových systémov spočíva v ich schopnosti integrovať sa s umelou inteligenciou a poskytovať prediktívne analýzy“. [1]

Táto schopnosť umožňuje inteligentným domácnostiam nielen reagovať na aktuálne podmienky, ale aj predvídať potreby svojich používateľov, čo prináša nové možnosti personalizácie a optimalizácie.

1. Loxone: Komplexné riadenie domácnosti

Loxone, ktorý je postavený na miniserveri, slúži ako centrálny uzol pre komunikáciu medzi zariadeniami a databázami. Zbiera dáta zo senzorov, ako sú detektory pohybu či teplotné senzory, a umožňuje ich spracovanie na účely automatizácie. Používatelia môžu jednoducho ovládať svoju domácnosť prostredníctvom aplikácie alebo webového rozhrania.

Príkladom môže byť automatizácia osvetlenia, ktorá na základe údajov o pohybe upraví svetelné podmienky v miestnosti. Miniserver navyše podporuje integráciu s rôznymi technológiami, ako sú Zigbee a Z-Wave, čo zvyšuje jeho škálovateľnosť a možnosti pripojenia. [13]

2. Aqara: Bezdrôtové riešenie pre inteligentné domácnosti

Aqara sa špecializuje na jednoduchú inštaláciu a bezdrôtovú komunikáciu pomocou Zigbee protokolu. Jej riešenia ukladajú dáta do cloudu, čo umožňuje vzdialený prístup a integráciu s populárnymi platformami, ako sú Apple HomeKit alebo Google Assistant. Tento systém je ideálny pre menšie domácnosti alebo miesta, kde nie je možná káblová inštalácia.

[14]

Tabulka 1 Loxone vs Aqara

Funkcia	Loxone	Aqara
Typ riadenia	Centrálny miniserver	Cloudové riešenie
Komunikácia	Káblková a bezdrôtová (Loxone Air)	Bezdrôtová (Zigbee)
Škálovateľnosť	Vysoká, modulárne rozšírenia	Obmedzená na Zigbee
Správa údajov	Lokálne uloženie	Cloudové uloženie

(zdroj: vlastný)

Integrácia databázových systémov s inteligentnými riadiacimi jednotkami, ako sú Loxone a Aqara, ponúka mnohé výhody:

- Efektívne rozhodovanie: Automatizácia na základe reálnych údajov zvyšuje pohodlie a znižuje náklady.
- Väčšia bezpečnosť: Lokálne uloženie dát minimalizuje riziko úniku informácií.
- Jednoduchá správa: Používateľ má k dispozícii jednotné rozhranie na ovládanie a monitorovanie všetkých zariadení.

Prepojenie databáz a riadiacich systémov vytvára základ pre budúce inteligentné domy, ktoré budú schopné predvídať potreby svojich obyvateľov a prispôsobovať sa ich požiadavkám v reálnom čase.

6 BEZPEČNOST A OCHRANA DÁT V DATABÁZOVÝCH SYSTÉMOCH

Bezpečnosť databáz je predovšetkým ochrana citlivých údajov a informácii uložených v databáze pred neoprávneným prístupom, zneužitím alebo zničením. Toto zahŕňa rôzne techniky a stratégie, ktoré zabezpečujú, že prístup k údajom majú len oprávnené osoby. Kľúčovým cieľom u bezpečnosti databáz je ochrana tajných údajov pred ich neoprávneným zverejnením, zachovanie integrity údajov tým, že zabráni ich neautorizovaným úpravám, či poškodeniu, a zabezpečeniu dostupnosti údajov pre oprávnených užívateľov, práve vtedy keď ich potrebujú. Rovnako dôležité je splniť právne a regulačné požiadavky týkajúce sa ochrany údajov a súkromia. Týmto spôsobom bezpečnosť databáz podporuje spoľahlivé a bezpečné používanie informácií. [15]

6.1 Techniky zabezpečenia dát

1. Fyzické zabezpečenie

Fyzické zabezpečenie databáz zahŕňa opatrenia na ochranu pred neoprávneným prístupom alebo krádežou prostredníctvom fyzických bariér. Patrí sem zabezpečenie serverových miestností, zavedenie prístupových kontrol do dátových centier a využitie monitorovacích systémov, ako sú kamery a alarmy.

Hlavný význam fyzického zabezpečenia spočíva v ochrane hardvéru pred fyzickým poškodením alebo odcudzením, čo zaisťuje jeho nepretržitú dostupnosť a integritu. Tieto opatrenia zároveň chránia servery a úložiská pred neoprávneným prístupom, čím zvyšujú celkovú bezpečnosť uložených údajov.

2. Sieťová bezpečnosť

Sieťová bezpečnosť sa zameriava na ochranu databázy pred neoprávneným prístupom cez sieť. Používa opatrenia ako firewall, enkrypcia a systémy detekcie prienikov.

Význam sieťovej bezpečnosti spočíva v tom, že zabezpečuje bezpečný prenos údajov cez sieť, čím zabraňuje ich zachyteniu alebo modifikácii neoprávnenými osobami. Taktiež chráni databázu pred externými útokmi, ako sú hacking alebo malware.

[15]

3. Kontrola prístupu

Kontrola prístupu je metóda, ktorá obmedzuje prístup k databáze len na oprávnených používateľov. Zahŕňa autentifikáciu, autorizáciu a účtovanie.

- Autentifikácia overuje identitu používateľa pomocou prihlasovacích údajov, ako sú používateľské mená a heslá.
- Autorizácia riadi, aké akcie môže používateľ vykonávať na databáze na základe jeho rolí alebo oprávnení.
- Účtovanie zaznamenáva a audituje všetky aktivity vykonané na databáze na zabezpečenie zodpovednosti a súladu s predpismi.

Kontrola prístupu je kľúčová, pretože zabezpečuje, že len oprávnené osoby môžu pristupovať k databáze a vykonávať konkrétne akcie, čím sa zabráňuje neoprávneným úpravám alebo vymazaniu údajov.

4. Šifrovanie údajov

Šifrovanie údajov je technika, ktorá chráni údaje uložené v databáze tým, že ich zašifruje pomocou šifrovacích algoritmov. Aj v prípade, že neoprávnená osoba získa prístup k šifrovaným údajom, nebude ich môcť prečítať ani použiť.

Dôležitosť šifrovania spočíva v ochrane citlivých údajov pred zneužitím, čím sa zabráňuje narušeniu bezpečnosti a neoprávnenému prístupu.

5. Auditovanie a protokolovanie

Auditovanie a protokolovanie sú dôležité metódy na monitorovanie a sledovanie všetkých akcií vykonaných na databáze. Zaznamenávajú rôzne aktivity, ako sú prihlásenia používateľov, úpravy údajov a systémové udalosti.

Tieto metódy sú nevyhnutné na odhalenie a prevenciu bezpečnostných incidentov. Zároveň pomáhajú splniť regulačné a legislatívne požiadavky týkajúce sa ochrany údajov a súkromia, čím prispievajú k lepšej bezpečnosti a zodpovednosti.

6.2 Hrozby pre bezpečnosť databáz

1. SQL injekcie

Táto forma útoku nastáva, keď útočník vloží škodlivý SQL kód do dotazu, aby prinútil databázu vykonávať neautorizované príkazy. To môže viesť k neoprávnenému prístupu k citlivým údajom, manipulácii s dátami alebo úplnému narušeniu bezpečnosti databázy. Výsledkom môže byť získanie, zmena alebo zneužitie informácií bez povolenia, čo umožňuje ďalšie útoky.

2. Neoprávnený prístup

Ide o situácie, keď jednotlivci získajú prístup k databáze bez oprávnenia. Môže sa to stať pomocou odcudzených prihlasovacích údajov, slabých autentifikačných mechanizmov alebo nesprávne nastavených prístupových práv. Takýto prístup často vedie k neoprávnenému zverejneniu, manipulácii alebo zneužitiu citlivých údajov.

3. Malvérové útoky

Malvér, ako vírusy, červy alebo ransomware, predstavuje vážnu hrozbu pre databázy. Tieto škodlivé programy sa môžu dostať do systému prostredníctvom infikovaných súborov alebo škodlivých odkazov. Môžu spôsobiť narušenie bezpečnosti údajov, poškodenie obsahu databázy alebo jej zneužitie na nelegálne účely.

4. Vnútorne hrozby

Tieto útoky pochádzajú od osôb, ktoré majú legitímny prístup k databáze, no tento prístup zneužijú na osobný prospech alebo spôsobenie škody. Patria sem neoprávnený prístup k údajom, ich krádež alebo pokusy o narušenie či poškodenie databázového systému.

5. DoS útoky

Cieľom DoS útokov je narušiť dostupnosť a funkčnosť databázy zaplavením systému množstvom požiadaviek alebo škodlivou prevádzkou. To môže spôsobiť nedostupnosť databázy, jej zrušenie a potenciálnu stratu údajov, čo vedie k prerušeniu služieb.

[15]

6.3 Osvedčené postupy pre zabezpečenie databáz

Firmy a organizácie, ktoré využívajú databázy na uchovávanie citlivých a dôverných informácií, musia zabezpečiť ich ochranu pred neoprávneným prístupom a možnými bezpečnostnými incidentmi. Preto je kľúčové implementovať spoľahlivé bezpečnostné opatrenia, ktoré zabezpečia ochranu údajov a pomôžu efektívne zvládať potenciálne hrozby.

1. Používanie silných hesiel

Heslá predstavujú prvú obrannú líniu proti neoprávnenému prístupu do databáz. Slabé heslá umožňujú hackerom jednoduchšie získať prístup k citlivým informáciám. Preto je dôležité používať silné heslá, ktoré sú zložité a ťažko odhadnuteľné. Takéto heslá by mali obsahovať kombináciu veľkých a malých písmen, čísiel a špeciálnych znakov.

Heslá by sa mali pravidelne meniť, najmä pri odchode zamestnanca zo spoločnosti, aby sa zabránilo používaniu starých prihlasovacích údajov. Okrem toho je vhodné zaviesť dvojfaktorovú autentifikáciu, ktorá pridáva ďalšiu vrstvu ochrany. Tento spôsob vyžaduje kombináciu dvoch typov overenia, napríklad heslo a bezpečnostný token alebo biometrickú autentifikáciu. Aj keď dôjde k odhaleniu hesla, dvojfaktorová autentifikácia výrazne znižuje riziko neoprávneného prístupu.

2. Obmedzenie prístupu

Obmedzenie prístupu k databázam je účinným spôsobom, ako zabrániť neoprávnenému prístupu k citlivým údajom. Nie všetci zamestnanci alebo používatelia potrebujú prístup ku všetkým údajom uloženým v databáze. Prístup by sa mal riadiť princípom minimálnych oprávnení, čo znamená poskytnutie len toho prístupu, ktorý je nevyhnutný na splnenie konkrétnej úlohy.

Napríklad, ak zamestnanec potrebuje prístup k údajom o zákazníkoch, mal by mať prístup iba k tejto časti databázy. Okrem toho je dôležité monitorovať a sledovať aktivity používateľov v databáze, aby sa mohli okamžite identifikovať a nahlásiť neoprávnené prístupy alebo podozrivé správanie.

3. Pravidelné aktualizácie a opravy

Obmedzenie prístupu k databázam je účinným spôsobom, ako zabrániť neoprávnenému prístupu k citlivým údajom. Nie všetci zamestnanci alebo používatelia potrebujú prístup ku všetkým údajom uloženým v databáze. Prístup by sa mal riadiť princípom minimálnych oprávnení, čo znamená poskytnutie len toho prístupu, ktorý je nevyhnutný na splnenie konkrétnej úlohy.

Napríklad, ak zamestnanec potrebuje prístup k údajom o zákazníkoch, mal by mať prístup iba k tejto časti databázy. Okrem toho je dôležité monitorovať a sledovať aktivity používateľov v databáze, aby sa mohli okamžite identifikovať a nahlásiť neoprávnené prístupy alebo podozrivé správanie.

4. Monitorovanie anomálií

Efektívne odhalenie a prevencia potenciálnych bezpečnostných hrozieb sa dá dosiahnuť monitorovaním anomálií v databáze. Detekcia anomálií zahŕňa neustále sledovanie aktivít v databáze na identifikáciu neobvyklého správania, ktoré sa odchyľuje od očakávaných noriem.

Napríklad náhly nárast prenosu dát alebo pokus o neoprávnený prístup k obmedzeným údajom môžu byť označené ako anomálne správanie. Tieto udalosti môžu spustiť opatrenia na vyšetrovanie a zmiernenie rizík. Na monitorovanie anomálií sa využívajú nástroje, ako sú systémy detekcie prienikov, analyzátory sieťovej prevádzky a nástroje na analýzu protokolov. Tieto technológie pomáhajú identifikovať potenciálne riziká a zavádzať potrebné opatrenia na ochranu údajov.

[15]

II PRAKTICKÁ ČÁST

7 NÁVRH RELAČNÝCH DATABÁZ PRE CHYTRÉ DOMÁCNOSTI

Relačné databázy zohrávajú kľúčovú úlohu pri implementácii inteligentných domácností. Ich robustná štruktúra, ktorá využíva tabuľky a logické vzťahy, umožňuje efektívnu organizáciu dát generovaných rôznymi zariadeniami, ako sú senzory, termostaty a bezpečnostné systémy. Ako uvádzajú Davidson a Moss vo svojej knihe, „návrh relačnej databázy pre inteligentné systémy vyžaduje jasné definovanie entít, ich atribútov a vzťahov, aby bolo možné dosiahnuť optimálnu funkcionálnu a efektívnu funkčnosť systému“ [11]

7.1 Návrh databázovej štruktúry

Pri návrhu relačnej databázy pre chytrú domácnosť je dôležité identifikovať hlavné entity a ich vzájomné prepojenia. Davidson a Moss zdôrazňujú, že základné tabuľky by mali zahŕňať:

1. Zariadenia:

- Obsahujú informácie o všetkých zariadeniach v domácnosti, ako sú osvetlenie, vykurovacie systémy a inteligentné zásuvky.
- Hlavné atribúty zahŕňajú: ID zariadenia, názov zariadenia, typ zariadenia, aktuálny stav a čas poslednej aktualizácie.

2. Senzory:

- Senzory poskytujú dáta, ako je teplota, vlhkosť alebo detekcia pohybu.
- Atribúty: ID senzora, názov senzora, typ senzora (napr. teplotný, pohybový), hodnota a čas odoslania dát.

3. Miestnosti:

- Tabuľka miestností umožňuje zoskupovanie zariadení a senzorov podľa ich fyzického umiestnenia v domácnosti.
- Atribúty: ID miestnosti, názov miestnosti a popis.

4. Udalosti:

- Ukladanie historických údajov o aktivitách zariadení, ako je zapnutie/vypnutie svetla alebo nastavenie termostatu.
- Atribúty: ID udalosti, typ udalosti, čas udalosti a ID zariadenia.

Relačné databázy využívajú prepojenia medzi tabuľkami prostredníctvom primárnych a cudzích kľúčov. Davidson a Moss uvádzajú, že „definovanie jasných vzťahov medzi entitami je základom dobre navrhutej databázy“ [11]

Napríklad:

- Tabuľka Zariadenia je prepojená s tabuľkou Miestnosti pomocou cudzieho kľúča ID miestnosti.
- Tabuľka Udalosti odkazuje na tabuľku Zariadenia pomocou atribútu ID zariadenia, čím zabezpečuje sledovanie histórie aktivít jednotlivých zariadení. [11]

7.2 Tvorba diagramu – ERD

Pri návrhu systému pre inteligentnú domácnosť je dôležité vytvoriť databázu, ktorá bude spoľahlivo ukladať všetky potrebné informácie. Ide napríklad o údaje zo senzorov, informácie o zariadeniach, používateľoch a ich právach, vykonaných akciách, alebo o bezpečnostných udalostiach. Aby všetko fungovalo, musela byť databáza navrhnutá tak, aby boli dáta medzi sebou prepojené a zároveň zrozumiteľne usporiadané.

Proces návrhu začal identifikáciou hlavných objektov systému, ako sú senzory, zariadenia, akcie, miestnosti, používateľské role a oprávnenia. Na základe týchto entít bol vytvorený entitno-relačný (ERD) diagram, ktorý zachytáva vzťahy medzi jednotlivými tabuľkami. Následne bola relačná schéma implementovaná v prostredí Microsoft SQL Server Management Studio, kde boli definované všetky tabuľky, primárne a cudzie kľúče, ako aj potrebné väzobné tabuľky pre zabezpečenie mnohonásobných vzťahov.

Pre vizualizáciu návrhu relačnej databázy bol v prostredí ERDPlus vytvorený entitno-relačný diagram. Následne bol však upravený aj v samostatnom programe SQL Server Management Studio (SSMS). Tento diagram slúži na znázornenie tabuliek, ich atribútov a vzťahov medzi nimi pomocou primárnych a cudzích kľúčov. Využitie diagramu pomáha k lepšiemu pochopeniu logickej štruktúry databázy a jej vzťahov.

Diagram zahŕňa nasledujúce kľúčové oblasti:

1. Používateľská správa a oprávnenia:

Entita Pouzivatel reprezentuje jednotlivých používateľov systému, ktorí sú priradení k entite Rola, čo určuje ich oprávnenia. Väzba medzi rolami a oprávneniami je riešená pomocou tabuliek Rola, Opravnenie a ich prepojovacej tabuľky Opravnenie role.

2. Zariadenia a senzory:

Každé zariadenie je reprezentované entitou Zariadenie, pričom môže byť priradené k miestnosti (Miestnost). Zariadenie môže byť súčasťou viacerých bezpečnostných hlásení (Zariadenie Hlasenie) alebo vykonávať rôzne akcie (Zariadenie Akcia). Na zaznamenávanie fyzikálnych hodnôt slúži entita Senzor, ktorá je typovo rozlíšená entitou Typ senzora a má svoju hodnotu v Senzor hodnota. Tá je navyše prepojená s jednotkou merania (Jednotka).

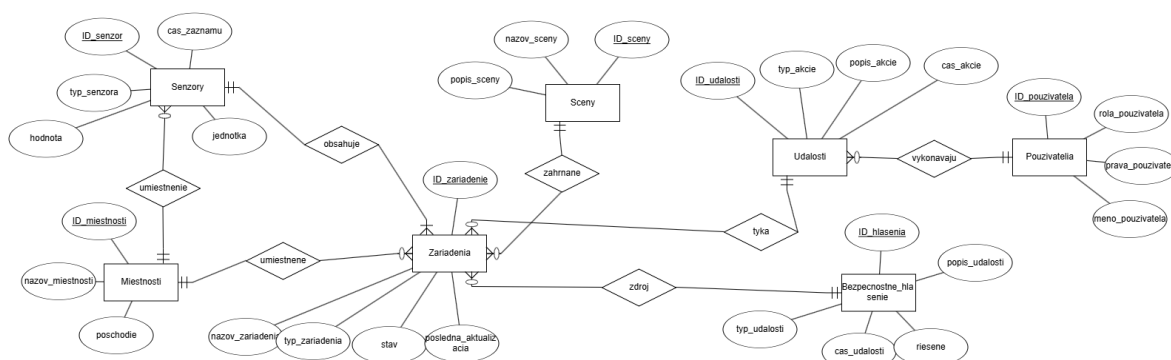
3. Bezpečnostné hlásenia:

Udalosti bezpečnostného charakteru sú evidované v entite Bezpečnostne Hlasenie, ktoré je spojené s konkrétnym zariadením (Zariadenie Hlasenie) a kategorizované podľa typu (Typ udalosti).

4. Scény a akcie:

Scény (Scena) predstavujú preddefinované sekvencie akcií, ktoré sú vykonávané pomocou zariadení. Spojenie medzi scénou a zariadením je riešené pomocou Zariadenie Scena. Samotné akcie sú definované v entite Akcia, ktorá je priradená k zariadeniu (Zariadenie Akcia) a typovo rozlíšená entitou Typ akcie.

Tento relačný model zabezpečuje prehľadné a štruktúrne oddelené uloženie informácií, ktoré reflektujú funkčnosti inteligentnej domácnosti. Prípadné úpravy alebo rozšírenia možno ľahko realizovať vďaka modularite modelu a použitiu väzbových tabuliek pre M:N vzťahy.



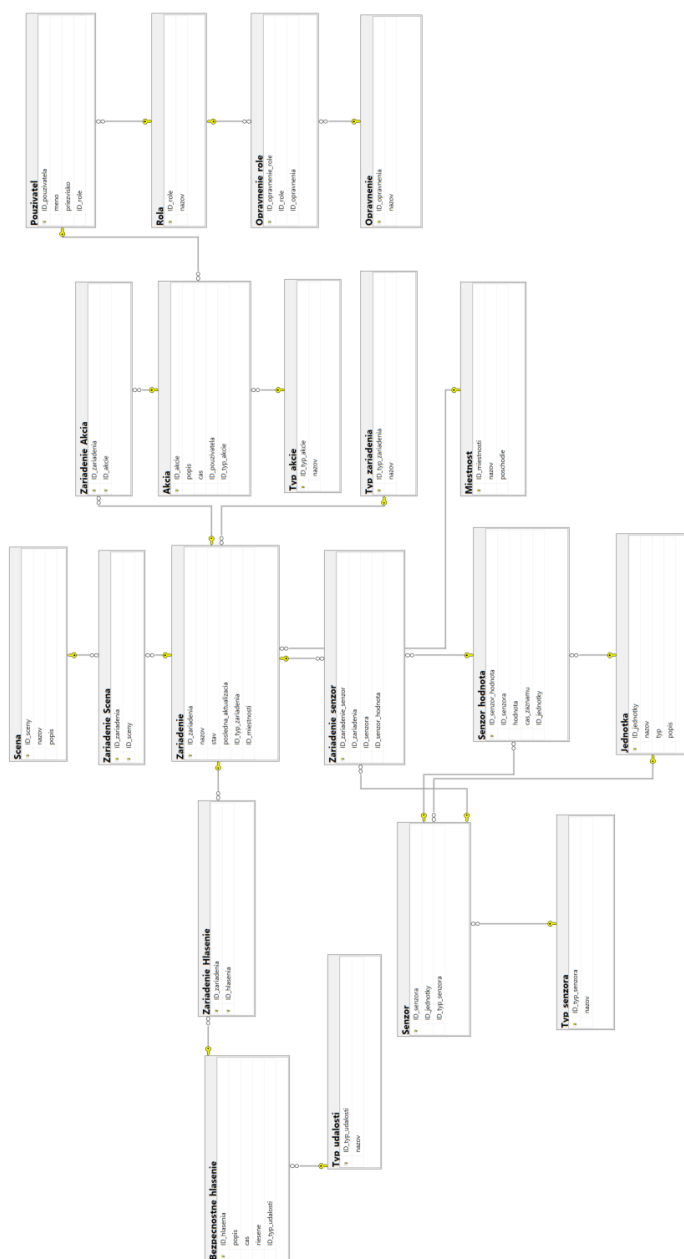
Obrázek 1 Prvotný návrh ERD

(zdroj: vlastný)

7.3 Tvorba relačnej schémy databázy

Z prvotného ER diagramu zobrazeného na Obrázku 1 bola postupnými úpravami vytvorená relačná schéma navrhovanej databázy (Obrázok 2).

Cieľom bolo mať databázu, ktorá je jasná, funkčná a pripravená aj na prípadné rozšírenia v budúcnosti. Týmto spôsobom vznikol základ, na ktorom môžu jednotlivé časti systému spolupracovať a vymieňať si informácie.



Obrázok 2 Relačná schéma databázy

(zdroj: vlastný)

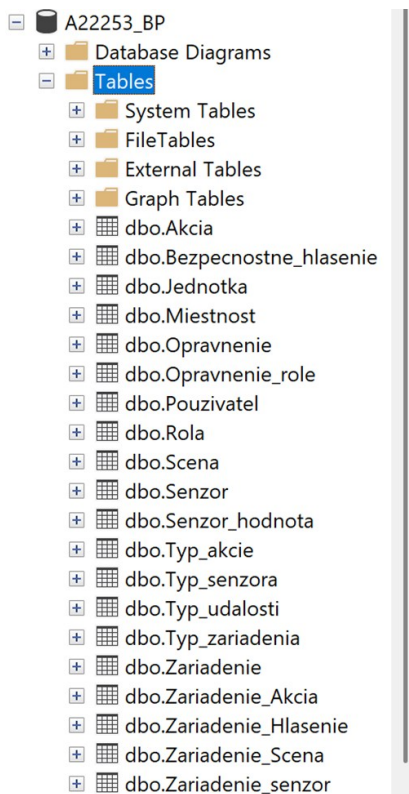
7.4 Tvorba tabuliek databázy

V tejto časti sa nachádzajú príkazy CREATE TABLE, ktoré slúžia na vytvorenie jednotlivých tabuliek databázy. Každá tabuľka obsahuje vhodne zvolené dátové typy, primárne a cudzie kľúče, ktoré zabezpečujú integritu údajov a správne väzby medzi tabuľkami.

```
CREATE TABLE Scena (  
    ID_sceny INT PRIMARY KEY,  
    nazov VARCHAR(100) NOT NULL,  
    popis TEXT NOT NULL);  
CREATE TABLE Jednotka (  
    ID_jednotky INT PRIMARY KEY,  
    nazov VARCHAR(50) NOT NULL,  
    popis VARCHAR(50) NULL,  
    typ VARCHAR(30) NULL);  
  
CREATE TABLE Typ_senzora (  
    ID_typ_senzora INT PRIMARY KEY,  
    nazov VARCHAR(50) NOT NULL);  
CREATE TABLE Typ_zariadenia (  
    ID_typ_zariadenia INT PRIMARY KEY,  
    nazov VARCHAR(50) NOT NULL);  
CREATE TABLE Typ_akcie (  
    ID_typ_akcie INT PRIMARY KEY,  
    nazov VARCHAR(50) NOT NULL);  
CREATE TABLE Typ_udalosti (  
    ID_typ_udalosti INT PRIMARY KEY,  
    nazov VARCHAR(50) NOT NULL);  
CREATE TABLE Opravnenie (  
    ID_opravnenia INT PRIMARY KEY,  
    nazov VARCHAR(50) NOT NULL);  
CREATE TABLE Rola (  
    ID_role INT PRIMARY KEY,  
    nazov VARCHAR(50) NOT NULL);  
CREATE TABLE Opravnenie_role (  
    ID_opravnenie_role INT PRIMARY KEY,  
    ID_role INT NOT NULL,  
    ID_opravnenia INT NOT NULL,  
    FOREIGN KEY (ID_role) REFERENCES Rola(ID_role),
```

```
FOREIGN KEY (ID_opravnenia) REFERENCES Opravnenie(ID_opravnenia));
CREATE TABLE Miestnost (
    ID_miestnosti INT PRIMARY KEY,
    nazov VARCHAR(100) NOT NULL,
    hodnota VARCHAR(50));
CREATE TABLE Senzor (
    ID_senzora INT PRIMARY KEY,
    ID_jednotky INT NOT NULL,
    ID_typ_senzora INT NOT NULL,
    FOREIGN KEY (ID_jednotky) REFERENCES Jednotka(ID_jednotky),
    FOREIGN KEY (ID_typ_senzora) REFERENCES Typ_senzora(ID_typ_senzora));
CREATE TABLE Pouzivatel (
    ID_pouzivatela INT PRIMARY KEY,
    meno VARCHAR(50) NOT NULL,
    priezvisko VARCHAR(50) NOT NULL,
    ID_role INT NOT NULL,
    FOREIGN KEY (ID_role) REFERENCES Rola(ID_role));
CREATE TABLE Akcia (
    ID_akcie INT PRIMARY KEY,
    popis TEXT NOT NULL,
    cas DATETIME NOT NULL,
    ID_pouzivatela INT NOT NULL,
    ID_typ_akcie INT NOT NULL,
    FOREIGN KEY (ID_pouzivatela) REFERENCES Pouzivatel(ID_pouzivatela),
    FOREIGN KEY (ID_typ_akcie) REFERENCES Typ_akcie(ID_typ_akcie));
CREATE TABLE Bezpecnostne_hlasenie (
    ID_hlasenia INT PRIMARY KEY,
    popis TEXT NOT NULL,
    cas DATETIME NOT NULL,
    riesene BIT NOT NULL,
    ID_typ_udalosti INT NOT NULL,
    FOREIGN KEY (ID_typ_udalosti) REFERENCES Typ_udalosti(ID_typ_udalosti));
CREATE TABLE Senzor_hodnota (
    ID_senzor_hodnota INT PRIMARY KEY,
    ID_senzora INT NOT NULL,
    hodnota FLOAT NOT NULL,
    cas_zaznamu DATETIME NOT NULL,
    FOREIGN KEY (ID_senzora) REFERENCES Senzor(ID_senzora));
CREATE TABLE Zariadenie (
    ID_zariadenia INT PRIMARY KEY,
```

```
nazov VARCHAR(100) NOT NULL,
stav VARCHAR(50),
posledna_aktualizacia DATETIME NOT NULL,
ID_typ_zariadenia INT NOT NULL,
ID_miestnosti INT NOT NULL,
FOREIGN KEY (ID_typ_zariadenia) REFERENCES
Typ_zariadenia(ID_typ_zariadenia),
FOREIGN KEY (ID_miestnosti) REFERENCES Miestnost(ID_miestnosti));
CREATE TABLE Zariadenie_senzor (
ID_zariadenie_senzor INT PRIMARY KEY,
ID_zariadenia INT NOT NULL,
ID_senzora INT NOT NULL,
ID_senzor_hodnota INT NOT NULL,
FOREIGN KEY (ID_zariadenia) REFERENCES Zariadenie(ID_zariadenia),
FOREIGN KEY (ID_senzora) REFERENCES Senzor(ID_senzora),
FOREIGN KEY (ID_senzor_hodnota) REFERENCES
Senzor_hodnota(ID_senzor_hodnota));
CREATE TABLE Zariadenie_Scena (
ID_zariadenia INT NOT NULL,
ID_sceny INT NOT NULL,
PRIMARY KEY (ID_zariadenia, ID_sceny),
FOREIGN KEY (ID_zariadenia) REFERENCES Zariadenie(ID_zariadenia),
FOREIGN KEY (ID_sceny) REFERENCES Scena(ID_sceny) );
CREATE TABLE Zariadenie_Akcia (
ID_zariadenia INT NOT NULL,
ID_akcie INT NOT NULL,
PRIMARY KEY (ID_zariadenia, ID_akcie),
FOREIGN KEY (ID_zariadenia) REFERENCES Zariadenie(ID_zariadenia),
FOREIGN KEY (ID_akcie) REFERENCES Akcia(ID_akcie));
CREATE TABLE Zariadenie_Hlasenie (
ID_zariadenia INT NOT NULL,
ID_hlasenia INT NOT NULL,
PRIMARY KEY (ID_zariadenia, ID_hlasenia),
FOREIGN KEY (ID_zariadenia) REFERENCES Zariadenie(ID_zariadenia),
FOREIGN KEY (ID_hlasenia) REFERENCES
Bezpecnostne_hlasenie(ID_hlasenia));
```



Obrázek 3 Zoznam vytvorených tabuliek v databáze

(zdroj: vlastný)

7.4.1 Tabuľka Zariadenie

Obsahuje zoznam všetkých zariadení použitých v systéme. Každé zariadenie má svoje ID, názov, stav (napr. zapnutý, vypnutý), dátum poslednej aktualizácie a odkazy na typ zariadenia, miestnosť, v ktorej sa nachádza, a priradené akcie, scény a hlásenia prostredníctvom väzbových tabuliek.

Pridanie údajov do tabuľky Zariadenie

```
INSERT INTO Zariadenie (ID_zariadenia, nazov, stav, posledna_aktualizacia,
ID_typ_zariadenia, ID_miestnosti) VALUES
(1, 'Loxone Touch Tree', 'zapnutý', '2025-05-06 18:02:27', 1, 1 ),
(2, 'Loxone Motion Sensor', 'aktívny', '2025-05-06 18:02:27', 4, 2 ),
(3, 'Loxone Temperature Sensor', 'zapnutý', '2025-05-06 18:02:27', 2, 3 ),
(4, 'Loxone Relay Extension', 'neaktívny', '2025-05-06 18:02:27', 3, 2),
(5, 'Loxone Smart Socket', 'aktívny', '2025-05-06 18:02:27', 5, 1 );
```

	ID_zariadenia	nazov	stav	posledna_aktualizacia	ID_typ_zariadenia	ID_miestnosti
1	1	Loxone Touch Tree	zapnutý	2025-05-06 18:02:27.000	1	1
2	2	Loxone Motion Sensor	aktívny	2025-05-06 18:02:27.000	4	2
3	3	Loxone Temperature Sensor	zapnutý	2025-05-06 18:02:27.000	2	3
4	4	Loxone Relay Extension	neaktívny	2025-05-06 18:02:27.000	3	2
5	5	Loxone Smart Socket	aktívny	2025-05-06 18:02:27.000	5	1

Obrázek 4 Tabuľka Zariadenie

(zdroj: vlastný)

7.4.2 Tabuľka Typ_zariadenia

Určuje kategórie zariadení (napr. osvetlenie, kúrenie, zabezpečovacie systémy). Umožňuje typizáciu a jednoduché filtrovanie alebo riadenie podobných zariadení.

```
INSERT INTO Typ_zariadenia (ID_typ_zariadenia, nazov) VALUES
(1, 'Osvetlenie'),
(2, 'Kúrenie'),
(3, 'Vetrание'),
(4, 'Zabezpečovací systém'),
(5, 'Multimédiá'),
(6, 'Žalúzie'),
(7, 'Zavlažovanie'),
(8, 'Prístupový systém');
```

	ID_typ_zariadenia	nazov
1	1	Osvetlenie
2	2	Kúrenie
3	3	Vetrание
4	4	Zabezpečovací systém
5	5	Multimédiá
6	6	Žalúzie
7	7	Zavlažovanie
8	8	Prístupový systém

Obrázek 5 Tabuľka Typ_zariadenia

(zdroj: vlastný)

7.4.3 Tabuľka miestnosť

Eviduje miestnosti v objekte (napr. kuchyňa, obývačka), v ktorých sa nachádzajú zariadenia. Tabuľka obsahuje názov a voliteľnú hodnotu ako napr. poschodie.

```
INSERT INTO Miestnost (ID_miestnosti, nazov, poschodie) VALUES
(1, 'Obývačka', '1. poschodie'),
(2, 'Kuchyňa', '1. poschodie'),
(3, 'Spálňa', '2. poschodie'),
(4, 'Chodba', '1. poschodie');
```

	ID_miestnosti	nazov	poschodie
1	1	Obývačka	1. poschodie
2	2	Kuchyňa	1. poschodie
3	3	Spálňa	2. poschodie
4	4	Chodba	1. poschodie

Obrázek 6 Tabuľka Miestnost

(zdroj: vlastný)

7.4.4 Tabuľka Scéna

Scény predstavujú preddefinované situácie, napr. "príchod domov" či "odchod z domu", ktoré ovplyvňujú viacero zariadení naraz. Scéna má svoj názov a popis.

```
INSERT INTO Scena (ID_sceny, nazov, popis) VALUES
(1, 'Nočný režim', 'Zhasnutie všetkých svetiel, zatiahnutie žalúzií,
aktivácia alarmu'),
(2, 'Ranné vstávanie', 'Rozviestenie svetiel v spálni, otvorenie žalúzií'),
(3, 'Odchod z domu', 'Vypnutie všetkých zariadení, aktivácia bezpečnostných
senzorov'),
(4, 'Príchod domov', 'Zapnutie svetiel v chodbe, odomknutie vstupných
dverí, deaktivácia alarmu'),
(5, 'Večera', 'Zapnutie teplého osvetlenia v kuchyni a jedálni'),
(6, 'Film', 'Stlmenie svetiel, zapnutie televízora a ozvučenia'),
(7, 'Vetranie', 'Otvorenie okien a deaktivácia kúrenia na určitý čas'),
(8, 'Nočná kúpeľňa', 'Zapnutie tlmeného osvetlenia v kúpeľni pri nočnom
pohybe')
```

	ID_sceny	nazov	popis
1	1	Nočný režim	Zhasnutie všetkých svetiel, zatiahnutie žalúzií, aktiv...
2	2	Ranné vstávanie	Rozviestenie svetiel v spálni, otvorenie žalúzií
3	3	Odchod z domu	Vypnutie všetkých zariadení, aktivácia bezpečnost...
4	4	Príchod domov	Zapnutie svetiel v chodbe, odomknutie vstupných ...
5	5	Večera	Zapnutie teplého osvetlenia v kuchyni a jedálni
6	6	Film	Stlmenie svetiel, zapnutie televízora a ozvučenia
7	7	Vetranie	Otvorenie okien a deaktivácia kúrenia na určitý čas
8	8	Nočná kúpeľňa	Zapnutie tlmeného osvetlenia v kúpeľni pri nočno...

Obrázek 7 Tabuľka Scéna

(zdroj: vlastný)

7.4.5 Tabuľka Zariadenie_Scena

Väzbová tabuľka medzi zariadeniami a scénami. Keďže jedno zariadenie môže patriť do viacerých scén a scéna môže obsahovať viacero zariadení, ide o vzťah typu M:N.

```
INSERT INTO Zariadenie_Scena (ID_zariadenia, ID_sceny) VALUES
(1, 1), -- Touch Tree v scéne "Príchod domov"
```

```
(2, 1), -- Motion Sensor tiež aktivovaný pri príchode domov
(3, 2), -- Temperature Sensor v scéne "Odchod z domu"
(4, 3), -- Relay Extension v scéne "Večerný režim"
(5, 4), -- Smart Socket v scéne "Film/TV režim"
(1, 5), -- Touch Tree aj v scéne "Ranné vstávanie"
(3, 5); -- Temperature Sensor pre ranné kúrenie
```

	ID_zariadenia	ID_sceny
1	1	1
2	1	5
3	2	1
4	3	2
5	3	5
6	4	3
7	5	4

Obrázek 8 Tabuľka Zariadenie_Scena

(zdroj: vlastný)

7.4.6 Tabuľka Akcia

Zaznamenáva vykonanie konkrétnej akcie (napr. zapnutie svetla). Obsahuje popis, čas, typ akcie a používateľa, ktorý ju vykonal.

```
INSERT INTO Akcia (ID_akcie,popis,ID_pouzivatela,ID_typ_akcie) VALUES
(1, 'Zapnutie osvetlenia v obývačke', 1, 1),
(2, 'Otvorenie garážových dverí', 2, 2);
(3, 'Zapnutie osvetlenia', GETDATE(), 1, 1),
(4, 'Detekcia pohybu', GETDATE(), 1, 2),
(5, 'Meranie teploty', GETDATE(), 2, 1),
(6, 'Vypnutie zásuvky', GETDATE(), 3, 3),
(7, 'Zapnutie relé', GETDATE(), 2, 4);
```

	ID_akcie	popis	cas	ID_pouzivatela	ID_typ_akcie
1	1	Zapnutie osvetlenia v obývačke	2025-05-06 19:35:52.987	1	1
2	2	Otvorenie garážových dverí	2025-05-06 19:35:52.987	2	2
3	3	Zapnutie osvetlenia	2025-05-06 20:07:44.970	1	1
4	4	Detekcia pohybu	2025-05-06 20:07:44.970	1	2
5	5	Meranie teploty	2025-05-06 20:07:44.970	2	1
6	6	Vypnutie zásuvky	2025-05-06 20:07:44.970	3	3
7	7	Zapnutie relé	2025-05-06 20:07:44.970	2	4

Obrázek 9 Tabuľka Akcia

(zdroj: vlastný)

7.4.7 Tabuľka Typ_akcie

Obsahuje zoznam typov akcií (napr. zapnutie, vypnutie, časované spustenie).

```
INSERT INTO Typ_akcie (ID_typ_akcie, nazov) VALUES
(1, 'Zapnutie zariadenia'),
(2, 'Vypnutie zariadenia'),
(3, 'Zmena teploty'),
(4, 'Otvorenie žalúzií'),
(5, 'Zatvorenie žalúzií'),
(6, 'Zaznamenanie pohybu'),
(7, 'Spustenie scény'),
(8, 'Zablokovanie vstupu');
```

	ID_typ_akcie	nazov
1	1	Zapnutie zariadenia
2	2	Vypnutie zariadenia
3	3	Zmena teploty
4	4	Otvorenie žalúzií
5	5	Zatvorenie žalúzií
6	6	Zaznamenanie pohybu
7	7	Spustenie scény
8	8	Zablokovanie vstupu

Obrázek 10 Tabuľka Typ_akcie

(zdroj: vlastný)

7.4.8 Tabuľka Zariadenie_Akcia

Väzbová tabuľka medzi zariadeniami a akciami (M:N). Jedno zariadenie môže byť predmetom viacerých akcií.

```
INSERT INTO Zariadenie_Akcia (ID_zariadenia, ID_akcie) VALUES
(1, 1), -- Loxone Touch Tree → zapnutie osvetlenia
(2, 2), -- Loxone Motion Sensor → detekcia pohybu
(3, 3), -- Temperature Sensor → spustenie kúrenia
(4, 4), -- Relay Extension → aktivácia vetrania
(5, 5), -- Smart Socket → zapnutie televízora
(1, 6), -- Touch Tree → deaktivácia svetla
(2, 6); -- Motion Sensor → vypnutie zabezpečenia
```

	ID_zariadenia	ID_akcie
1	1	1
2	1	6
3	2	2
4	2	6
5	3	3
6	4	4
7	5	5

Obrázek 11 Tabuľka Zariadenie_Akcia

(zdroj: vlastný)

7.4.9 Tabuľka Bezpecnostne_hlasenie

Zaznamenáva bezpečnostné incidenty, ako napr. vniknutie do objektu. Obsahuje popis, čas a stav riešenia.

```
INSERT INTO Bezpecnostne_hlasenie (ID_hlasenia, popis, cas, riesene,
ID_typ_udalosti) VALUES
(1, 'Systém zaznamenal neoprávnený vstup do objektu cez hlavné dvere',
GETDATE(), 0, 1),
(2, 'Hasiaci senzor zaznamenal výskyt dymu v kuchyni', GETDATE(), 1, 2),
(3, 'Senzor úniku vody detegoval zaplavenie technickej miestnosti',
GETDATE(), 0, 3),
(4, 'Teplota prekročila nastavený limit v serverovni', GETDATE(), 1, 4),
(5, 'Zariadenie sa odpojilo a prestalo odpovedať', GETDATE(), 0, 5),
(6, 'Používateľ ručne vypol zabezpečenie v nočnom režime', GETDATE(), 1,
6),
(7, 'Systém hlási stratu spojenia s vonkajšou kamerou', GETDATE(), 0, 7),
(8, 'Zaznamenaná zmena stavu systému - aktivácia režimu „Dovolenka“',
GETDATE(), 1, 8);
```

	ID_hlasenia	popis	cas	riesene	ID_typ_udalosti
1	1	Systém zaznamenal neoprávnený vstup do objektu c...	2025-05-06 19:41:19.583	0	1
2	2	Hasiaci senzor zaznamenal výskyt dymu v kuchyni	2025-05-06 19:41:19.583	1	2
3	3	Senzor úniku vody detegoval zaplavenie technickej ...	2025-05-06 19:41:19.583	0	3
4	4	Teplota prekročila nastavený limit v serverovni	2025-05-06 19:41:19.583	1	4
5	5	Zariadenie sa odpojilo a prestalo odpovedať	2025-05-06 19:41:19.583	0	5
6	6	Používateľ ručne vypol zabezpečenie v nočnom reži...	2025-05-06 19:41:19.583	1	6
7	7	Systém hlási stratu spojenia s vonkajšou kamerou	2025-05-06 19:41:19.583	0	7
8	8	Zaznamenaná zmena stavu systému - aktivácia režim...	2025-05-06 19:41:19.583	1	8

Obrázek 12 Tabuľka Bezpecnostne_hlasenie

(zdroj: vlastný)

7.4.10 Tabuľka Typ_udalosti

Obsahuje kategórie udalostí, ktoré sú dôvodom hlásení, napr. požiaru alebo únik vody.

```
INSERT INTO Typ_udalosti (ID_typ_udalosti, nazov) VALUES
(1, 'Vniknutie do objektu'),
(2, 'Detekcia požiaru'),
(3, 'Únik vody'),
(4, 'Prekročenie teplotného limitu'),
(5, 'Zlyhanie zariadenia'),
(6, 'Manuálny zásah užívateľa'),
(7, 'Strata spojenia so zariadením'),
(8, 'Zmena stavu systému');
```

	ID_typ_udalosti	nazov
1	1	Vniknutie do objektu
2	2	Detekcia požiaru
3	3	Únik vody
4	4	Prekročenie teplotného limitu
5	5	Zlyhanie zariadenia
6	6	Manuálny zásah užívateľa
7	7	Strata spojenia so zariadením
8	8	Zmena stavu systému

Obrázek 13 Tabuľka Typ_udalosti

(zdroj: vlastný)

7.4.11 Tabuľka Zariadenie_Hlasenie

Väzbová tabuľka medzi zariadeniami a hláseniami (M:N).

```
INSERT INTO Zariadenie_Hlasenie (ID_zariadenia, ID_hlasenia) VALUES
(1, 1), -- Loxone Touch Tree → Vniknutie do objektu
(2, 2), -- Motion Sensor → Detekcia požiaru
(3, 3), -- Temperature Sensor → Únik vody
(4, 4), -- Relay Extension → Prekročenie teplotného limitu
(5, 5), -- Smart Socket → Zlyhanie zariadenia
(3, 6), -- Temperature Sensor → Manuálny zásah užívateľa
(2, 7); -- Motion Sensor → Strata spojenia so zariadením
```

	ID_zariadenia	ID_hlasenia
1	1	1
2	2	2
3	2	7
4	3	3
5	3	6
6	4	4
7	5	5

Obrázek 14 Tabuľka Zariadenie_Hlasenie

(zdroj: vlastný)

7.4.12 Tabuľka Používateľ

Obsahuje zoznam používateľov, ktorí majú prístup do systému. Každý používateľ má meno, priezvisko a je priradený k roli.

```
INSERT INTO Pouzivatel (ID_pouzivatela, meno, priezvisko, ID_role)
VALUES
(1, 'Emma', 'Šlesariková', 1),    -- Administrátor
(2, 'Ján', 'Kováč', 3),           -- Používateľ
(3, 'Marek', 'Novotný', 2),       -- Technik
(4, 'Lucia', 'Kováčová', 4);      -- Host'
```

	ID_pouzivatela	meno	priezvisko	ID_role
1	1	Emma	Šlesariková	1
2	2	Ján	Kováč	3
3	3	Marek	Novotný	2
4	4	Lucia	Kováčová	4

Obrázek 15 Tabuľka Používateľ

(zdroj: vlastný)

7.4.13 Tabuľka Rola

Definuje oprávnenia používateľov v systéme, napr. administrátor, technik, host'.

```
INSERT INTO Rola (ID_role, nazov) VALUES
(1, 'Administrátor'),
(2, 'Technik'),
(3, 'Používateľ'),
(4, 'Host');
```

	ID_role	nazov
1	1	Administrátor
2	2	Technik
3	3	Použivateľ
4	4	Hosť

Obrázek 16 Tabuľka Rola

(zdroj: vlastný)

7.4.14 Tabuľka Oprávnenie

Zoznam jednotlivých práv (napr. zobrazenie zariadení, editovanie akcií).

```
INSERT INTO Opravnenie (ID_opravnenia, nazov) VALUES
(1, 'Zobraziť stav zariadení'),
(2, 'Ovládať zariadenia'),
(3, 'Pridávať nové zariadenia'),
(4, 'Odstraňovať zariadenia'),
(5, 'Zobrazovať záznamy zo senzorov'),
(6, 'Spravovať používateľov'),
(7, 'Zobrazovať bezpečnostné hlásenia'),
(8, 'Potvrdiť vyriešenie udalosti');
```

	ID_opravnenia	nazov
1	1	Zobraziť stav zariadení
2	2	Ovládať zariadenia
3	3	Pridávať nové zariadenia
4	4	Odstraňovať zariadenia
5	5	Zobrazovať záznamy zo senzorov
6	6	Spravovať používateľov
7	7	Zobrazovať bezpečnostné hlásenia
8	8	Potvrdiť vyriešenie udalosti

Obrázek 17 Tabuľka Oprávnenie

(zdroj: vlastný)

7.4.15 Tabuľka Opravnenie_role

Väzba medzi rolami a oprávneniami (M:N), kde každá rola môže mať viaceré oprávnenia.

```
INSERT INTO Opravnenie_role (ID_opravnenie_role, ID_role,
ID_opravnenia) VALUES
(1, 1, 1), -- zobrazenie
(2, 1, 2), -- úprava
(3, 1, 3), -- mazanie
```

```

(4, 1, 4); -- pridavanie
-- Priradenie oprávnení pre Technika (ID_role = 2)
INSERT INTO Opravnenie_role (ID_opravnenie_role, ID_role,
ID_opravnenia)
VALUES
(5, 2, 1), -- zobrazenie
(6, 2, 2), -- úprava
-- Priradenie oprávnení pre Používateľa (ID_role = 3)
INSERT INTO Opravnenie_role (ID_opravnenie_role, ID_role,
ID_opravnenia)
VALUES
(7, 3, 1); -- zobrazenie
-- Host (ID_role = 4) nemá žiadne oprávnenia, alebo napr. len
zobrazenie:
INSERT INTO Opravnenie_role (ID_opravnenie_role, ID_role,
ID_opravnenia)
VALUES
(8, 4, 1); -- zobrazenie

```

	ID_opravnenie_role	ID_role	ID_opravnenia
1	1	1	1
2	2	1	2
3	3	1	3
4	4	1	4
5	5	2	1
6	6	2	2
7	7	3	1
8	8	4	1

Obrázek 18 Tabuľka Opravnenie_role

(zdroj: vlastný)

7.4.16 Tabuľka Senzor

Zaznamenáva senzory (napr. teplotný senzor) a ich priradenie k typu a jednotke.

```

INSERT INTO Senzor (ID_senzora, ID_jednotky, ID_typ_senzora) VALUES
(1, 1, 1), -- Teplotný senzor → °C
(2, 2, 2), -- Senzor vlhkosti → %
(3, 3, 3), -- Pohybový senzor → boolean
(4, 6, 4), -- Dverový kontakt → otvorené/zatvorené
(5, 4, 5), -- CO2 senzor → ppm
(6, 5, 6), -- Svetelný senzor → lux
(7, 1, 7), -- Vodeodolný teplotný senzor → °C
(8, 7, 8); -- Multifunkčný senzor → vlastná jednotka

```

	ID_senzora	ID_jednotky	ID_typ_senzora
1	1	1	1
2	2	2	2
3	3	3	3
4	4	6	4
5	5	4	5
6	6	5	6
7	7	1	7
8	8	7	8

Obrázek 19 Tabuľka Senzor

(zdroj: vlastný)

7.4.17 Tabuľka Typ_senzora

Obsahuje typy senzorov (napr. teplotný, pohybový, CO2).

```
INSERT INTO Typ_senzora (ID_typ_senzora, nazov) VALUES
(1, 'Teplotný senzor'),
(2, 'Senzor vlhkosti'),
(3, 'Pohybový senzor'),
(4, 'Dverový kontakt'),
(5, 'CO2 senzor'),
(6, 'Svetelný senzor'),
(7, 'Vodeodolný teplotný senzor'),
(8, 'Multifunkčný senzor');
```

	ID_typ_senzora	nazov
1	1	Teplotný senzor
2	2	Senzor vlhkosti
3	3	Pohybový senzor
4	4	Dverový kontakt
5	5	CO2 senzor
6	6	Svetelný senzor
7	7	Vodeodolný teplotný senzor
8	8	Multifunkčný senzor

Obrázek 20 Tabuľka Typ_senzora

(zdroj: vlastný)

7.4.18 Tabuľka Jednotka

Určuje jednotky merania (napr. °C, %, ppm).

```
INSERT INTO Jednotka (ID_jednotky, nazov, popis, typ) VALUES
(1, '°C', 'Teplota v stupňoch Celzia', 'teplota'),
(2, '%', 'Relatívna vlhkosť vzduchu', 'vlhkosť'),
```

```
(3, 'lux', 'Intenzita osvetlenia', 'svetlo'),
(4, 'ppm', 'Koncentrácia látky v častiach na milión', 'kvalita
ovzdušia'),
(5, 'kWh', 'Spotreba energie', 'energia'),
(6, 'otvorené/zatvorené', 'Stav kontaktu - dvere/okno', 'Poloha'),
(7, 'iné', 'Špeciálne alebo kombinované jednotky', 'Iné');
```

	ID_jednotky	nazov	typ	popis
1	1	°C	teplota	Teplota v stupňoch Celzia
2	2	%	vlhkosť	Relatívna vlhkosť vzduchu
3	3	lux	svetlo	Intenzita osvetlenia
4	4	ppm	kvalita ovzdušia	Koncentrácia látky v častiach na milión
5	5	kWh	energia	Spotreba energie
6	6	otvorené/zatvorené	Poloha	Stav kontaktu – dvere/okno
7	7	iné	Iné	Špeciálne alebo kombinované jednotky

Obrázek 21 Tabuľka Jednotka

(zdroj: vlastný)

7.4.19 Tabuľka Senzor_hodnota

Ukladá historické hodnoty meraní senzorov spolu s časom zaznamenania.

```
INSERT INTO Senzor_hodnota (ID_senzor_hodnota, ID_senzora, hodnota,
cas_zaznamu, ID_jednotky) VALUES
(1, 1, 22.5, GETDATE(), 1), -- °C
(2, 2, 45.2, GETDATE(), 2), -- %
(3, 3, 1, GETDATE(), 3), -- lux
(4, 4, 0, GETDATE(), 4), -- ppm
(5, 1, 23.0, GETDATE(), 1), -- °C
(6, 5, 0.04, GETDATE(), 4), -- ppm
(7, 6, 180.0, GETDATE(), 3); -- lux
```

	ID_senzor_hodnota	ID_senzora	hodnota	cas_zaznamu	ID_jednotky
1	1	1	22,5	2025-05-06 19:59:45.627	1
2	2	2	45,2	2025-05-06 19:59:45.627	2
3	3	3	1	2025-05-06 19:59:45.627	3
4	4	4	0	2025-05-06 19:59:45.627	4
5	5	1	23	2025-05-06 19:59:45.627	1
6	6	5	0,04	2025-05-06 19:59:45.627	4
7	7	6	180	2025-05-06 19:59:45.627	3

Obrázek 22 Tabuľka Senzor_hodnota

(zdroj: vlastný)

7.4.20 Tabuľka Zariadenie_senzor

Väzbová tabuľka medzi zariadením, senzorom a jeho hodnotou. Umožňuje sledovanie špecifických hodnôt na základe činnosti zariadenia.

```
INSERT INTO Zariadenie_senzor (ID_zariadenie_senzor, ID_zariadenia,
ID_senzora, ID_senzor_hodnota) VALUES
(1, 1, 1, 1), -- Loxone Touch Tree má Teplotný senzor s hodnotou 22.5
(2, 2, 2, 2), -- Motion Sensor má senzor vlhkosti
(3, 3, 3, 3), -- Temperature Sensor má pohybový senzor (ukážka
kombinácie)
(4, 4, 4, 4), -- Relay Extension má dverový kontakt
(5, 1, 1, 5), -- Loxone Touch Tree má opäť Teplotný senzor s novou
hodnotou
(6, 5, 5, 6), -- Smart Socket má CO2 senzor
(7, 2, 6, 7); -- Motion Sensor má aj svetelný senzor
```

	ID_zariadenie_senzor	ID_zariadenia	ID_senzora	ID_senzor_hodnota
1	1	1	1	1
2	2	2	2	2
3	3	3	3	3
4	4	4	4	4
5	5	1	1	5
6	6	5	5	6
7	7	2	6	7

Obrázek 23 Tabuľka Zariadenie_senzor

(zdroj: vlastný)

8 ZABEZPEČENIE DATABÁZY

Pri tvorbe databázového systému je tiež dôležité myslieť nielen na zabezpečenie štruktúry dát ale aj na ich ochranu pred neoprávneným prístupom. Jedným z hlavných spôsobov zabezpečenia je riadenie prístupu prostredníctvom používateľských účtov, ktoré definujú aké operácie môže daný používateľ v databáze vykonávať.

8.1 Používateľské účty a prístupy v databázovom systéme

V prostredí SQL server management studio (SSMS) je možné vytvoriť nového používateľa. V sekcii „security“ a následne v „logins“ je možné pridať nový login. Tento login môže, byť autentifikovaný buď prostredníctvom Windows Authentication, alebo SQL Server Authentication, kde používateľ zadáva vlastné meno a heslo. Po vytvorení loginu je možné použiť záložku „user mapping“ na pridanie databázy a špecifikáciu rolí, ktoré používateľovi určujú jeho oprávnenia.

Rola predstavuje súbor oprávnení, ktoré môžu byť priradené jednému alebo viacerým používateľom. SQL server poskytuje niekoľko preddefinovaných databázových rolí:

- Db_owner – úplný prístup k databáze, vrátane možnosti meniť štruktúru databázy a nastavovať oprávnenia.
- Db_datareader – používateľ má oprávnenie len na čítanie údajov zo všetkých tabuliek a pohľadov
- Db_datawriter – umožňuje vkladanie, aktualizáciu a mazanie údajov vo všetkých tabuľkách databázy
- Db_ddladmin – môže vytvárať a meniť objekty v databáze (napr. tabuľky, pohľady).
- Public – základná rola, ktorú má automaticky každý používateľ. Zvyčajne nemá špecifické práva.

Príkladom jednoduchého nastavenia môže byť vytvorenie používateľa „EmmaUser“ ktorému bude udelený len prístup na čítanie údajov.

```
Create login EmmaUser with password = 'Emma2025!';  
Use A22253_BP;  
Create user EmmaUser for login EmmaUser;  
Exec sp_addrolemember 'db_datareader', 'EmmaUser';
```

Týmto spôsobom bude mať používateľ prístup k celej databáze, no len na úrovni čítania.

8.2 Zabezpečenie tabuliek a dát prostredníctvom oprávnení

Zabezpečenie databázy sa neobmedzuje iba na správu používateľov a rolí. Dôležitou súčasťou ochrany dát je kontrola oprávnení na úrovni jednotlivých tabuliek, pohľadov, funkcií. Tento mechanizmus sa v SQL nazýva ako oprávnenia.

Príkladom bola vytvorená rola pre používateľa, ktorý má prístup len k údajom senzorov, ale nie k osobným údajom iných používateľov. Pomocou SQL príkazov bolo toto správanie zabezpečené nasledovne:

```
Deny select on dbo.Pouzivatel to EmmaUser;  
Grant select on dbo.Senzor to EmmaUser;
```

Týmto spôsobom má používateľ „EmmaUser“ možnosť zobrazit' informácie o senzore, ale nie je mu možné čítať údaje z tabuľky „Pouzivatel“, kde sú uložené osobné informácie.

Správne nakonfigurované oprávnenia znižujú riziko náhodného alebo úmyselného poškodenia dát, zamedzujú neoprávnenému prístupu k citlivým informáciám a pomáhajú databázovému systému spĺňať legislatívne požiadavky, napríklad nariadenie GDPR. Zároveň umožňujú efektívne rozdeliť zodpovednosti medzi jednotlivých používateľov systému, čím prispievajú k vyššej bezpečnosti a spoľahlivosti celého riešenia.

9 LOXONE CONFIG

9.1 Rozdelenie miestnosti a ich konfigurácia

Loxone config je oficiálny softvér určený na konfiguráciu a programovanie zariadení v systéme inteligentnej domácnosti Loxone. Slúži ako hlavný nástroj na vytváranie logiky, nastavovanie pravidiel, prepojenie hardvérových komponentov a optimalizovanie chodu domácnosti podľa potrieb užívateľa.

Program nám umožňuje pracovať s funkčnými blokmi, ktoré predstavujú základné prvky systému, napríklad ovládanie osvetlenia, kúrenia, tienia, zabezpečenia alebo multimédií. Vďaka tomu je možné jednoduché prepojenie medzi snímačmi, výstupnými zariadeniami a pokročilou logikou.

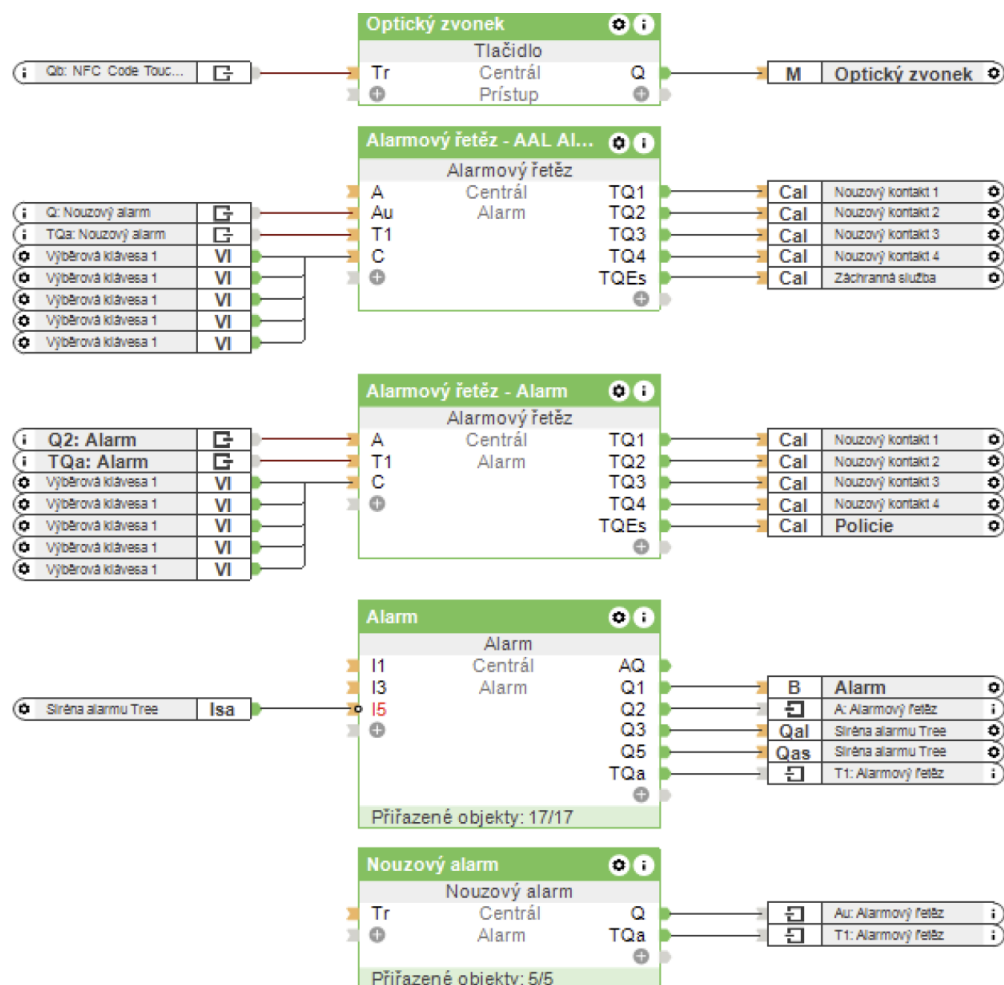
Veľkou výhodou Loxone config je možnosť využiť automatické programovanie, ktoré na základe dostupných zariadení a ich priradenia k miestnostiam dokáže vygenerovať základnú logiku automatizácie. Používateľ si ju následne môže upraviť a rozšíriť o ďalšie vlastné funkcie.

Loxone config podporuje prácu s viacerými typmi Extension modulov a technológií. Čo umožňuje široké možnosti nasadenia v rámci domov, bytov alebo aj menších komerčných objektov. Programovanie prebieha offline a následne sa konfigurácia nahrá do centrálného miniservera, ktorý následne riadi celý systém.

Celkovo je tento program robustný, ale zároveň aj prehľadný nástroj, ktorý umožňuje aj pokročilým používateľom navrhovať automatizačné systémy bez potreby znalosti klasického programovania.

9.1.1 Centrálny

Slúži ako hlavný riadiaci bod systému. V tomto prípade plní viacero úloh súvisiacich s bezpečnosťou a signalizáciou.



Obrázek 24 Centrál - Loxone Config

(zdroj: vlastní)

NFC Code touch a optický zvonček

NFC code touch slouží ku kontrole přístupu a je vybavený čítačkou NFC a působí jako přístupový bod. Použivatelé mohou zadat kód nebo využít NFC čip na vstup do objektu. Výstup z NFC code touch zařízení (např. Výstup Qb) může v rámci plnit více funkcí. Podle naprogramované logiky dokáže spustit alarmový řetězec, aktivovat optický zvonček při dveřích, nebo sloužit na spuštění či deaktivaci alarmu podle zadaného kódu.

Blok optický zvonček je speciálně určený na aktivaci světelného výstupu. Typicky LED světla nebo blesku, při stlačení NFC tlačítka. Funguje jako tichý zvonček, což je ideálně kdežto akustický zvuk může rušit ostatních lidí. Vizuální signalizace poskytuje spolehlivou informaci o přítomnosti návštěvy.

1. Alarmový reťazec – AAL reťazec

Alarmový reťazec predstavuje pokročilý funkčný blok určený na riadenie eskalácie incidentov v rámci zabezpečovacieho systému. Jeho hlavnou úlohou je zabezpečiť postupné a automatizované vykonávanie akcií po zistení narušenia objektu alebo iného definovaného incidentu. Blok obsahuje niekoľko vstupov a výstupov, ktoré umožňujú flexibilné riadenie celého priebehu alarmovej situácie. V stup označený ako „A“ slúži na aktiváciu celého reťazca, teda spustenie preddefinovaných krokov reakcie systému. Vstup T1 slúži ako hlavný trigger (spúšťač) – teda ak sa spustí napríklad alarm alebo je zaznamenaný neoprávnený vstup, tento vstup aktivuje reťazovú reakciu. Reťazec následne spustí výstupy TQ1 až TQ5, ktoré sú určené na rôzne výstupné akcie. Tie môžu byť napríklad aktivácia sirény, blikajúceho svetla, ale aj automatizované odoslanie upozornení či volaní cez telefónny výstup (napr. Cal). V reálnom nasadení môže byť takto zabezpečené napríklad volanie záchranej služby, upozornenie polície alebo odoslanie správy vlastníčkovi nehnuteľnosti. Tento blok je preto kľúčovým prvkom pri návrhu inteligentného zabezpečovacieho systému.

2. Alarmový reťazec – Alarm

Alarm predstavuje ďalší funkčný blok určený na automatickú reakciu systému pri výskyte bezpečnostnej udalosti. Hoci funguje na rovnakom princípe ako predchádzajúci reťazec AAL (Alarmový reťazec), slúži ako samostatná logická vetva. Čo umožňuje rozdeliť jednotlivé incidenty podľa závažnosti alebo typu zóny. Vďaka samostatnému bloku môže každý typ incidentu spúšťať odlišnú reakciu, napríklad tichý poplach pri panike alebo hlasitý výstražný systém režim pri vlamaní. Vstup T1 opäť slúži ako spúšťač celého reťazca, teda aktivuje výstupy a ďalšie následné akcie. Zásadný význam má však aj vstup C – „Clear“ – ktorý slúži na zrušenie alebo reset reťazca. Tento vstup sa typicky využíva pri úspešnom overení identity, deaktivácii alarmu oprávnenou osobou alebo v prípade potreby ručného ukončenia incidentu.

3. Blok Alarm

Predstavuje to centrálny prvok zabezpečovacej logiky v systéme Loxone. Jeho úlohou je monitorovanie vstupov zo senzorov a následná reakcia na neautorizované udalosti, ako sú pohyb, otvorenia okna či dverí, alebo idý podnet, ktorý signalizuje

narušení objektu. Vstupy bloku sú typicky označené ako I1, I3, I5, každý z nich je pripojený na konkrétny senzor alebo zónu (napríklad pohybový senzor v chodbe, magnetický kontakt na okne, alebo detektor prítomnosti). Aktiváciou ktoréhokoľvek z týchto vstupov dôjde k spusteniu alarmu, ak je systém aktívny.

Blok má viacero výstupov:

- AQ, Q1, Q2, Q3: logické výstupy, ktoré sa môžu použiť na aktiváciu svetelných signalizácií, notifikácií, záznam do logu alebo ďalšie funkcie podľa potreby.
- TQa, T1: špeciálne výstupy určené na napojenie do alarmových reťazcov – tým sa zabezpečí eskalácia incidentu (napr. spustenie sirény, notifikácie na mobil, aktivácia externého výstupu).
- Siréna: konkrétny výstup, ktorý je pripojený na Sirénu Tree – tá sa aktivuje akustickým signálom v prípade poplachu.
- Qas: vizualizačný výstup – poskytuje spätnú väzbu o stave alarmu do vizualizácie alebo aplikácie (napr. zobrazenie „Alarm aktívny“ alebo „Zóna narušená“).

4. Siréna tree

Siréna Tree je výstupné zariadenie určené na akustickú signalizáciu poplachu v inteligentnom zabezpečovacom systéme. Zariadenie je pripojené prostredníctvom Tree technológie, čo zabezpečuje rýchlu a spoľahlivú komunikáciu s Miniserverom. Ovládanie sirény je zabezpečené výstupom Q3 z bloku „Alarm“. To znamená, že ak dôjde k aktivácii alarmu na základe vstupov zo senzorov výstup Q3 sa aktivuje a okamžite spustí zvukový signál zo sirény. Siréna Tree je dôležitým prvkom zabezpečenia, slúži nielen na vyplašenie narušiteľa, ale aj na upozornenie obyvateľov alebo okolia, že došlo k poplachu.

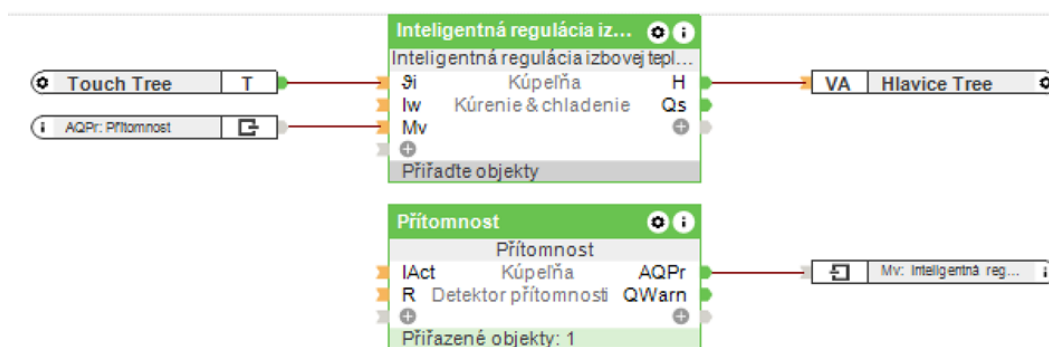
5. Núdzový alarm

Predstavuje dôležitý bezpečnostný prvok inteligentného systému, ktorý umožňuje rýchlu manuálnu aktiváciu alarmu v prípade núdze. Využíva sa najmä v situáciách, kedy používateľ potrebuje okamžite upozorniť na vzniknutý problém, typicky ide o tzv. „panické tlačidlo“, ktoré môže byť fyzicky umiestnené na dostupnom mieste alebo zabudované do ovládacieho panelu.

9.1.2 Kúpeľňa

V kúpeľni je implementovaná jednoduchá, no efektívna logika inteligentného vykurovania, ktorá reaguje na prítomnosť osoby v miestnosti. Základom tejto logiky je detektor prítomnosti Tree, ktorý priebežne monitoruje pohyb a zaznamenáva, či sa v priestore niekto nachádza. Ak senzor zistí prítomnosť, odošle signál do bloku „Inteligentná regulácia izby“ čím dôjde k aktivácii kúrenia. Paralelne je v miestnosti umiestnený aj dotykový vypínač (napr. Touch Tree), ktorý slúži na manuálne prepínanie režimov kúrenia. Používateľ si tak môže zvoliť komfortný režim, úsporu energie alebo kúrenie úplne vypnúť. Tento ovládací prvok ponúka flexibilitu v prípade, že používateľ nechce byť odkázaný len na automatiku.

Celá logika je navrhnutá tak, aby zabezpečila tepelný komfort iba vtedy, keď je to potrebné, čím sa šetrí energia a zároveň sa zvyšuje pohodlie používateľov.



Obrázek 25 Kúpeľňa - Loxone Config

(zdroj: vlastný)

1. Blok „Inteligentná regulácia izby“

Blok „Inteligentná regulácia izby“ je v systéme Loxone zodpovedný za automatické riadenie kúrenia a chladenia v kúpeľni. Jeho cieľom je udržiavať požadovanú teplotu v priestore na základe aktuálnych podmienok a používateľských preferencií.

Blok má viacero vstupov, ktoré ovplyvňujú jeho správanie:

- Si – manuálny vstup, do ktorého je pripojený výstup z dotykového ovládača. Tento vstup umožňuje používateľovi prepínať medzi rôznymi režimami kúrenia, ako sú „Komfort“, „Úspora“ alebo „Vypnuté“.
- Iw – vstup pre externý zdroj kúrenia alebo chladenia. V tomto prípade je len označený, ale nie je aktívne využitý.

- Mv – vstup z bloku „Prítomnosť“, ktorý indikuje, či sa v miestnosti niekto nachádza. Ak áno, systém prejde do aktívneho režimu vykurovania.

Výstupy z bloku zabezpečujú riadenie fyzických zariadení:

- H – hlavný teplotný výstup, ktorý riadi vykurovacie zariadenie, konkrétne Hlavicu Tree. Tá na základe tohto signálu otvára alebo zatvára prietok vody do radiátora alebo podlahového kúrenia.
- Qs – pomocný signalizačný výstup, ktorý môže byť využitý na vizuálne alebo logické indikovanie stavu kúrenia (napr. či je kúrenie aktívne alebo v útlme).

Táto logika zabezpečuje, že kúrenie v kúpeľni sa aktivuje len vtedy, keď je v miestnosti zaznamenaná prítomnosť osoby alebo keď to používateľ manuálne zapne. Výsledkom je úsporné a zároveň komfortné riadenie teploty, ktoré sa prispôsobuje skutočnému využívaniu priestoru.

2. Blok „Prítomnosť“

Blok „Prítomnosť“ v systéme Loxone slúži na vyhodnocovanie aktuálneho stavu obsadenosti miestnosti. Jeho primárnou úlohou je identifikovať, či sa v priestore nachádza osoba, a túto informáciu ďalej využiť pri riadení iných funkcií, ako sú osvetlenie, vykurovanie alebo bezpečnostné prvky.

Do bloku vstupujú dva vstupy:

- R (Ruch) – hlavný vstup z detektora prítomnosti (napr. AQR Tree), ktorý poskytuje informáciu o fyzickom pohybe v miestnosti.
- IAct – voliteľný manuálny vstup, ktorý môže byť použitý na ručné aktivovanie prítomnosti (v tomto prípade nie je využitý).

Blok generuje niekoľko výstupov:

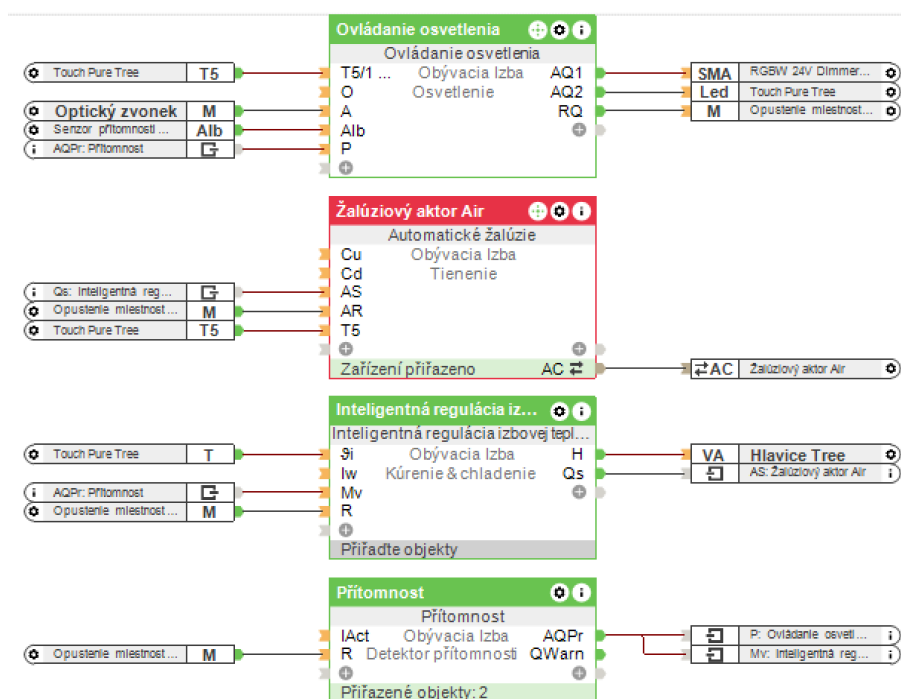
- AQPr – aktívna prítomnosť; výstup je nastavený na logickú hodnotu 1, ak je v miestnosti detegovaný pohyb.
- QWarn – výstražný signál, ktorý indikuje potenciálnu chybu v detekcii (napríklad ak bola očakávaná prítomnosť, ale nebola potvrdená).

V konkrétnom prípade kúpeľne je výstup AQPr prepojený na vstup Mv bloku „Inteligentná regulácia izby“. To znamená, že systém zohľadňuje prítomnosť osoby pri rozhodovaní, či má byť kúrenie aktivované. V prípade, že sa v miestnosti nikto

nenachádza, kúrenie ostáva vypnuté alebo prejde do úsporného režimu. Tento princíp prispieva k zníženiu spotreby energie a zároveň zabezpečuje komfort iba vtedy, keď je to potrebné.

9.1.3 Obývacia izba

Je centrálné miesto pobytu v domácnosti zohráva kľúčovú úlohu aj v rámci inteligentného riadenia. Jej automatizácia je zameraná najmä na zabezpečenie pohodlia, energetickej efektivity a zároveň flexibilita ovládania. V rámci riešenia boli do obývačky implementované viaceré prvky inteligentného systému, od osvetlenia a tienenia, cez riadenie teploty až po detekciu prítomnosti. Všetky tieto funkcie sú navzájom prepojené a pracujú v súlade, čím sa zabezpečuje optimálny komfort pri minimálnej potrebe manuálnych zásahov zo strany používateľa.



Obrázek 26 Obývací izba - Loxone Config

(zdroj: vlastný)

1. Ovládanie osvetlenia

Osvetlenie v obývacej izbe je riadené pomocou bloku „Ovládanie osvetlenia“, ktorý zabezpečuje automatické aj manuálne ovládanie svetelných zón v závislosti od rôznych vstupov. Tento blok predstavuje centrálny prvok pre správu svetelných scenárov, stmievania a automatických reakcií na prítomnosť.

Na vstupe bloku sa nachádza niekoľko signálov:

- T5/1 je manuálny impulz z dotykového panela Touch Pure Tree, pomocou ktorého môže používateľ zapnúť alebo vypnúť osvetlenie.
- Alb je vstup z detektora prítomnosti, ktorý slúži na automatické zapnutie osvetlenia pri vstupe do miestnosti.
- M je vstup z logiky opustenia miestnosti, keď systém vyhodnotí, že sa v priestore nikto nenachádza, odošle signál na automatické zhasnutie svetiel.

Výstupy bloku slúžia na ovládanie konkrétnych svetelných okruhov:

- AQ1 a AQ2 sú výstupy pre dve svetelné zóny, napríklad hlavné stropné osvetlenie a doplnkové ambientné svetlo.
- RQ poskytuje stavový výstup logiky, ktorý môže byť použitý na vizualizáciu alebo ďalšie logické väzby.

K výstupom sú pripojené konkrétne zariadenia:

- SMA predstavuje výstup na RGBW 24V Dimmer Tree, ktorý zabezpečuje stmievanie a farebné nastavenie osvetlenia.
- Led ovláda podsvietenie ikoniek na dotykovom paneli Touch Pure Tree.
- M je zároveň výstup, ktorý sa aktivuje pri opustení miestnosti, čím zhasne všetky svetelné zdroje pre úsporu energie.

Táto logika zabezpečuje komfortné osvetlenie v obývačke, ktoré sa prispôbuje aktuálnej prítomnosti, umožňuje manuálne ovládanie a zároveň pracuje energeticky efektívne bez zbytočných zásahov.

2. Žalúziový aktér Air

V rámci inteligentného riadenia obývačky je súčasťou systému aj ovládanie tieniacej techniky – konkrétne žalúzií. Túto funkciu zabezpečuje blok „Žalúziový aktor“, ktorý umožňuje plne automatické aj manuálne ovládanie žalúzií podľa potreby používateľa alebo podmienok v miestnosti.

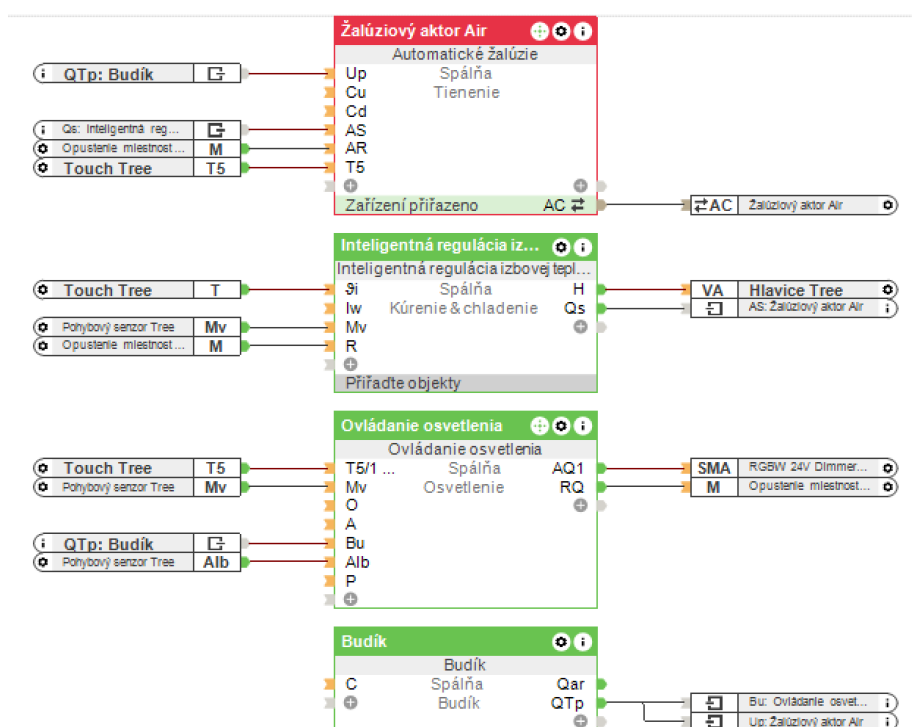
Blok prijíma viacero vstupov:

- Cu (Curtain up) a Cd (Curtain down) slúžia na manuálne vytiahnutie alebo spustenie žalúzií.

- AS a AR sú vstupy pre automatické stiahnutie alebo vytiahnutie, často sa používajú pri nastavení scén alebo podľa režimu miestnosti (napr. budenie alebo odchod).
- T5 je impulzný vstup z dotykového panela Touch Tree, ktorý umožňuje používateľovi ovládať žalúzie priamo dotykom.
- Na výstupe sa nachádza signál AC, ktorý je prepojený so zariadením Žalúziový aktor Tree. Toto hardvérové zariadenie následne fyzicky ovláda motor žalúzií.

9.1.4 Spáľňa

Spáľňa je jednou z najdôležitejších miestností v domácnosti z pohľadu komfortu, relaxácie a osobného súkromia. V inteligentnej domácnosti má preto mimoriadne postavenie, automatizácia v tejto miestnosti je navrhnutá tak, aby podporovala kvalitný spánok, pohodlné prebúdžanie a zároveň zabezpečila energetickú efektívnosť. Riešenie spálne v rámci systému Loxone zahŕňa viacero prepojených funkčných celkov: automatické osvetlenie, riadenie vykurovania, detekciu prítomnosti, ovládanie žalúzií a voliteľne aj inteligentný budík, ktorý umožňuje šetrné zobudenie pomocou svetla a prirodzeného svetelného rytmu.



Obrázek 27 Spáľňa - Loxone Config

(zdroj: vlastný)

1. Budík

Funkcia budíka v Loxone tak poskytuje moderný spôsob prebúdzania, ktorý nielenže nahrádza nepríjemný zvuk klasického alarmu, ale zároveň umožňuje naprogramovať budenie podľa režimu domácnosti, pracovných dní alebo individuálnych preferencií členov domácnosti. V kombinácii s osvetlením a žalúziami vytvára pre používateľa príjemné a prirodzené prostredie pri prechode z nočného do denného režimu.

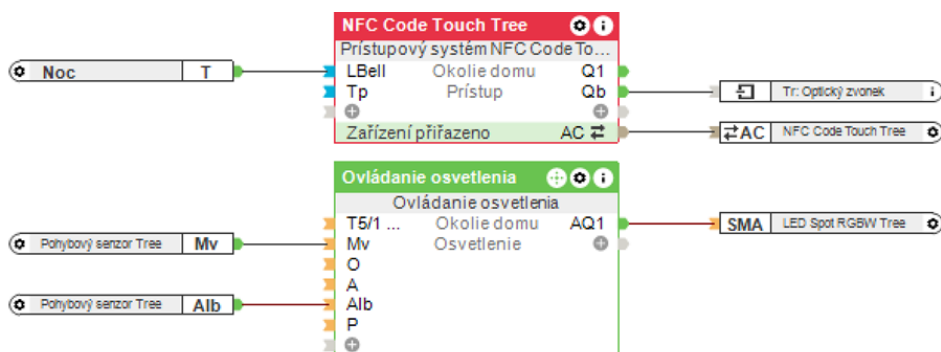
Vstupom do bloku je signál T, ktorý pochádza z výstupu QTP – ten môže byť generovaný napríklad časovačom alebo iným manuálnym/automatickým spínačom. Tento vstup určuje, kedy má budík prebehnúť – teda v aký čas sa má používateľ začať prebúdzat'.

Výstupy:

- Qar – slúži na ovládanie osvetlenia. Tento výstup je pripojený na vstup Bu (budenie) v bloku Ovládanie osvetlenia, kde spôsobí jemné a postupné rozsvietenie svetiel. Tým dochádza k simulácii prirodzeného svitania.
- QTP – tento výstup je pripojený na vstup Up bloku Žalúzie, čím sa pri budíku automaticky vytiahnu žalúzie a do miestnosti začne prenikať denné svetlo.

9.1.5 Okolie domu

V rámci systému Loxone bolo navrhnuté tak, aby zabezpečovalo základné funkcie ako riadenie prístupu, osvetlenie priestorov pred vstupom a notifikácie v prípade príchodu osoby k objektu. Logika je postavená na prepojení niekoľkých komponentov: NFC Code Touch, pohybového senzora, optického zvončeka a inteligentného osvetlenia.



Obrázek 28 Okolie domu - Loxone Config

(zdroj: vlastný)

1. Režim „Noc“

Režim „Noc“ v systéme Loxone slouží na změnu správy celého systému počas nočných hodín alebo v čase, keď si používatelia želajú pokojový režim. Po jeho aktivácii sa systém správa inak ako počas dňa – so zameraním na minimalizáciu rušenia a zvýšenie bezpečnosti.

V rámci logiky okolia domu je režim „Noc“ pripojený ako vstup T do zariadenia NFC Code Touch Tree. Aktivácia tohto vstupu môže spôsobiť zmenu viacerých prvkov systému:

- Zvonček môže prejsť do tichého režimu, teda aktivuje sa len optický výstup bez zvukového signálu (napr. len blikajúce svetlo namiesto gongu),
- Osvetlenie v exteriéri môže byť nastavené na tlmenú intenzitu alebo sa zapína len na kratší čas pri pohybe,
- Prístup cez NFC Code Touch môže byť blokovaný alebo obmedzený,

V prípade aktivácie alarmu môže systém reagovať diskretnejšie, napr. bez aktivácie vonkajšej sirény.

9.2 Cvičenia pre študentov

9.2.1 Automatické osvetlenie chodby pomocou pohybového senzora

Nakonfigurovať v prostredí Loxone Config, logiku, ktorá automaticky zapne svetlo v chodbe pri detekcii pohybu a vypne ho po určitom čase, ak nie je žiadna aktivita – Úloha 1.

Potrebné bloky:

- Pohybový senzor tree
- Ovládanie osvetlenia
- Výstup pre svetlo

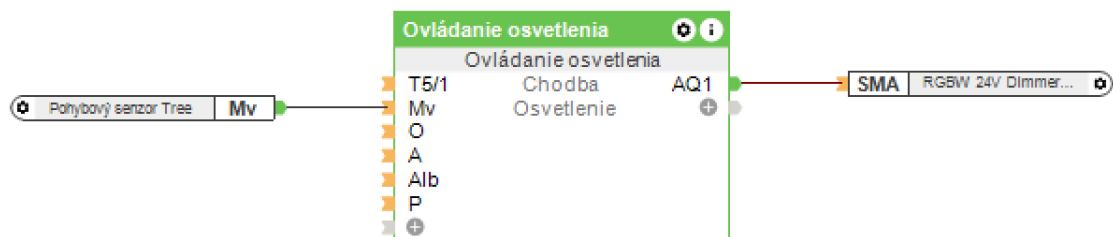
Postup:

1. Pridať zariadenie do miestnosti „Chodba“:

- Pohybový senzor
- Výstup na svetlo (RGBW dimmer)

2. Vložiť blok „ovládanie osvetlenia“

3. Pripoj výstup z pohybového senzora
 - Výstup Mv z pohybového senzora pripoj do vstupu Mv bloku osvetlenia.
4. Pripoj výstup bloku osvetlenia (napr. AQ1) k aktoru svetla.
5. Nastav čas vypnutia:
 - V nastaveniach osvetlenia nastav parameter čas vypnutia napr. na 2 minúty (čas bez pohybu po ktorom sa svetlo vypne).



Obrázek 29 Úloha 1 - Loxone Config

(zdroj: vlastný)

9.2.2 Aktivovať zabezpečovací režim

Nakonfigurovať v Loxone config blok „alarm“ tak, aby sa pri aktivácii režimu 'Odchod z domu' systém automaticky prepol do stráženého stavu po uplynutí 10-sekundovej oneskorenej doby – Úloha 2.

Potrebné bloky:

- Alarm
- Digitálny vstup (odchod z domu)
- Digitálny výstup (siréna)

Postup:

1. Digitálny vstup pre „Odchod z domu“ pripojíme na vstup V v bloku Alarm, ktorý slúži na aktiváciu režimu oneskoreného zastrešenia – teda aby sa alarm aktivoval až po krátkom odpočítavaní. Tento režim umožňuje používateľovi opustiť domácnosť bez toho, aby sa alarm spustil okamžite.

2. Využijeme výstup AQr (alebo Aqr, podľa verzie), ktorý zobrazuje zostávajúci čas do aktivácie alarmu. Tento výstup môžeme napojiť na vizualizáciu, ak chceme, aby používateľ videl, koľko sekúnd zostáva do zapnutia zabezpečenia.
3. V parametroch bloku nastavíme čas oneskorenia na 10 sekúnd (z nastavených 60 sekúnd), čím zabezpečíme rýchlejšiu aktiváciu systému.
4. Po uplynutí tejto doby výstup AQ z bloku Alarm nadobudne hodnotu 1, čo indikuje, že systém je plne zastrešený a pripravený reagovať na narušenie (napr. pohyb, otvorenie okna atď.)



Obrázek 30 Úloha 2 - Loxone Config

(zdroj: vlastný)

9.2.3 Spustenie alarmu pri narušení priestoru

Nakonfigurovať alarmový systém tak, aby sa spustil po detekcii pohybu alebo otvorení okna – Úloha 3.

Potrebné bloky:

- Pohybový senzor tree
- Magnetický kontakt
- Siréna tree
- Digitálne vstupy

Postup:

1. Pridanie vstupov:
 - Pripojíme pohybový senzor na digitálny vstup označený ako I1.
 - Pripojíme magnetický kontakt na vstup I3.
 - Oba vstupy vedú do bloku Alarm, kde aktivujú stav narušenia.

2. Nastavenie režimu aktivácie:

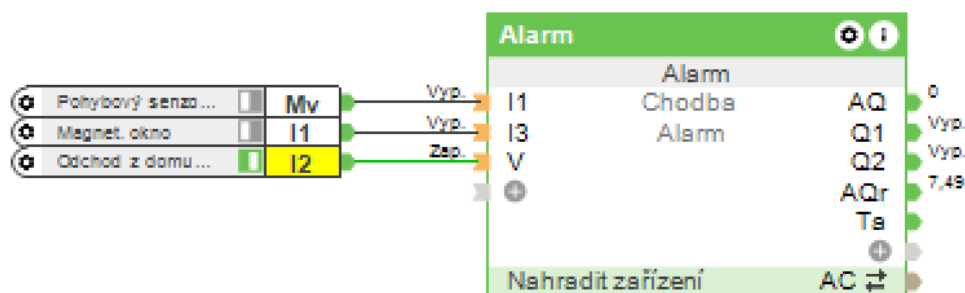
- Vstup V (oneskorené zastrešenie) slúži na to, aby sa alarm spustil napr. 10 sekúnd po odchode z domu.
- Tento vstup je napojený na tlačidlo alebo logiku "Odchod z domu".

3. Výstupy z alarmu:

- Q1 – aktivuje tichý alarm (napr. push notifikácia alebo vizuálna indikácia),
- Q2 – aktivuje sirénu Tree – akustický alarm.

4. Signalizácia stavu:

- Výstup AQr indikuje zostávajúci čas pred úplnou aktiváciou alarmu,
- Výstup AQ značí, že alarm je aktívny.



Obrázek 31 Úloha 3 - Loxone Config

(zdroj: vlastný)

9.2.4 Prístupový systém s rozpoznáním používatele

Rozpoznanie správneho kódu používateľa pomocou NFC Code touch zariadenia – Úloha 4.

Potrebné bloky:

- NFC Code touch tree
- RGBW Dimmer tree
- Oneskorenie vypnutia
- Ovládanie osvetlenia

Postup:

1. NFC Code Touch Tree

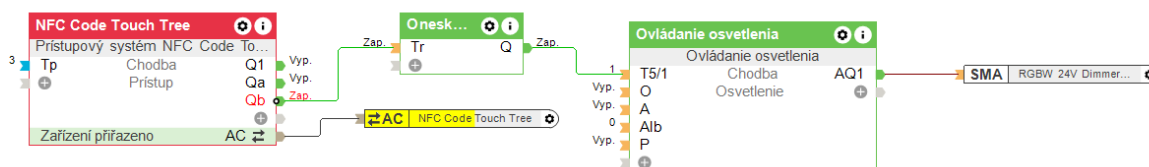
- Výstup Qb → aktivuje sa pri správnom kóde → ide na vstup Tr bloku „Oneskorenie vypnutia“.

2. Oneskorenie vypnutia

- Vstup Tr z NFC bloku,
- Výstup Q ide na vstup T5/1 v bloku „Ovládanie osvetlenia“,
- V parametroch nastavíme Tx (čas vypnutia) – napr. 5 sekúnd.

3. Ovládanie osvetlenia

- Vstup T5/1 z časovača,
- Výstup AQ1 → ide na výstupné zariadenie (napr. SMA = RGBW Dimmer Tree)



Obrázek 32 Úloha 4 - Loxone Config

(zdroj: vlastný)

9.2.5 Zariadenia priradené ku konkrétnej scéne (Príchod domov)

Výpis zariadenia priradeného ku konkrétnej scéne – Úloha 5

```
SELECT z.nazov AS Zariadenie
FROM Zariadenie z
JOIN Zariadenie_Scena zs ON z.ID_zariadenia = zs.ID_zariadenia
JOIN Scena s ON zs.ID_sceny = s.ID_sceny
WHERE s.nazov = 'Príchod domov';
```

	Zariadenie
1	Loxone Smart Socket

Obrázek 33 Úloha 5 – SSMS

(zdroj: vlastný)

9.2.6 Bezpečnostné hlásenia, ku ktorým sa viažu zariadenia

Výpis bezpečnostných hlásení, ku ktorým sa viažu zariadenia – Úloha 6

```
SELECT z.nazov AS Zariadenie, b.popis AS Hlasenie, b.cas
FROM Zariadenie z
JOIN Zariadenie_Hlasenie zh ON z.ID_zariadenia = zh.ID_zariadenia
JOIN Bezpecnostne_hlasenie b ON zh.ID_hlasenia = b.ID_hlasenia;
```

	Zariadenie	Hlasenie	cas
1	Loxone Touch Tree	Systém zaznamenal neoprávnený vstup do objektu c...	2025-05-06 19:41:19.583
2	Loxone Motion Sensor	Hasiaci senzor zaznamenal výskyt dymu v kuchyni	2025-05-06 19:41:19.583
3	Loxone Motion Sensor	Systém hlási stratu spojenia s vonkajšou kamerou	2025-05-06 19:41:19.583
4	Loxone Temperature Sensor	Senzor úniku vody detegoval zaplavenie technickej ...	2025-05-06 19:41:19.583
5	Loxone Temperature Sensor	Používateľ ručne vypol zabezpečenie v nočnom reži...	2025-05-06 19:41:19.583
6	Loxone Relay Extension	Teplota prekročila nastavený limit v serverovni	2025-05-06 19:41:19.583
7	Loxone Smart Socket	Zariadenie sa odpojilo a prestalo odpovedať	2025-05-06 19:41:19.583

Obrázek 34 Úloha 6 – SSMS

(zdroj: vlastný)

9.2.7 Zobrazenie všetkých zariadení v jednotlivých miestnostiach s ich typom a aktuálnym stavom

Výpis všetkých zariadení v jednotlivých miestnostiach s ich typom a aktuálnym stavom – Úloha 7

```
SELECT
    Z.nazov AS Zariadenie,
    M.nazov AS Miestnost,
    T.nazov AS Typ_zariadenia,
    Z.stav,
    Z.posledna_aktualizacia
FROM Zariadenie Z
JOIN Miestnost M ON Z.ID_miestnosti = M.ID_miestnosti
JOIN Typ_zariadenia T ON Z.ID_typ_zariadenia = T.ID_typ_zariadenia
ORDER BY M.nazov, Z.nazov;
```

	Zariadenie	Miestnost	Typ_zariadenia	stav	posledna_aktualizacia
1	Loxone Motion Sensor	Kuchyňa	Zabezpečovací systém	aktívny	2025-05-06 18:02:27.000
2	Loxone Relay Extension	Kuchyňa	Vetranie	neaktívny	2025-05-06 18:02:27.000
3	Loxone Smart Socket	Obývačka	Multimédiá	aktívny	2025-05-06 18:02:27.000
4	Loxone Touch Tree	Obývačka	Osvetlenie	zapnutý	2025-05-06 18:02:27.000
5	Loxone Temperature Sensor	Spálňa	Kúrenie	zapnutý	2025-05-06 18:02:27.000

Obrázek 35 Úloha 7 – SSMS

(zdroj: vlastný)

9.2.8 Prehľad zaznamenaných hodnôt senzorov vrátane typu senzora, jednotky času a merania

Výpis zaznamenaných hodnôt senzorov vrátane typu senzora, jednotky času a merania – Úloha 8

```
SELECT
    Sensor_hodnota.hodnota,
    Sensor_hodnota.cas_zaznamu,
    Typ_senzora.nazov AS Typ_senzora,
    Jednotka.nazov AS Jednotka
FROM Sensor_hodnota
JOIN Sensor ON Sensor_hodnota.ID_senzora = Sensor.ID_senzora
JOIN Typ_senzora ON Sensor.ID_typ_senzora = Typ_senzora.ID_typ_senzora
JOIN Jednotka ON Sensor.ID_jednotky = Jednotka.ID_jednotky
ORDER BY Sensor_hodnota.cas_zaznamu DESC;
```

	hodnota	cas_zaznamu	Typ_senzora	Jednotka
1	22,5	2025-05-06 19:59:45.627	Teplotný senzor	°C
2	45,2	2025-05-06 19:59:45.627	Senzor vlhkosti	%
3	1	2025-05-06 19:59:45.627	Pohybový senzor	lux
4	0	2025-05-06 19:59:45.627	Dverový kontakt	otvorené/zatvorené
5	23	2025-05-06 19:59:45.627	Teplotný senzor	°C
6	0,04	2025-05-06 19:59:45.627	CO2 senzor	ppm
7	180	2025-05-06 19:59:45.627	Svetelný senzor	kWh

Obrázek 36 Úloha 8 – SSMS

(zdroj: vlastný)

ZÁVER

Databázové systémy sú v dnešnej dobe neoddeliteľnou súčasťou inteligentných domácností. Umožňujú spracovávať a zhromažďovať údaje zo senzorov a zariadení, ktoré pomáhajú automatizovať rôzne činnosti, zvyšovať pohodlie obyvateľov a šetriť čas aj energiu.

V teoretickej časti bakalárskej práce boli predstavené a popísané základné princípy fungovania inteligentných domácností a ich spojitosť s databázovými systémami. Boli tu tiež popísané postupy týkajúce sa návrhu databáz a na používanie SQL dotazov, ktoré umožňujú jednoduchú prácu s uloženými údajmi. Dôležitou súčasťou bolo aj preskúmanie technológií, ako sú Loxone a Aqara, ktoré ukazujú, ako databázy spolupracujú s riadiacimi systémami na ovládanie domácností.

V praktickej časti bakalárskej práce bol predstavený návrh relačnej databáze pre ukladanie dát potrebných pre inteligentnú domácnosť, bol vytvorený ER diagram a z neho následne relačné schéma. Tabuľky relačnej schémy boli implementované do MS SQL Serveru a pre vyskúšanie naplnené ukázkovými údajmi. Pre predstavenie technológie Loxone študentom, prípadne širšej verejnosti boli tiež pripravené ukázkové scvičenia pre študentov, v ktorých bolo vytvorených 8 ukázkových úloh.

Časť bakalárskej práce bola tiež venovaná bezpečnosti údajov, čo je dôležité vzhľadom na množstvo citlivých informácií, ktoré tieto systémy spracovávajú. Táto práca ponúka prehľad základných poznatkov a pripravuje pôdu pre ďalšie praktické kroky, konkrétne realizáciu databázového systému pre chytrú domácnosť.

ZOZNAM POUŽITEJ LITERATURY

- [1] AUGUSTO, Juan Carlos a NUGENT, Chris D. (ed.). *Designing smart homes: the role of artificial intelligence*. Lecture notes in computer science. Berlin: Springer, c2006. ISBN 9783540359944.
- [2] EL-AZAB, Rasha. *Smart homes: potentials and challenges*. Clean Energy. 2021, s. 14.
- [3] *Čo je inteligentná domácnosť a 4 rady ako začať*. Online. KOTI. Dostupné z: <https://koti.sk/clanky/navody/co-je-inteligentna-domacnost-smart-home/>. [cit. 2025-01-03].
- [4] *Čo je to databáza ?* Online. Oracle. Dostupné z: <https://www.oracle.com/sk/database/what-is-database/>. [cit. 2025-01-03].
- [5] *10 Importance & Benefits of Database Management System*. Online. RalanTech. Dostupné z: https://www.ralantech.com/resources/10-advantages-of-database-management-system/?utm_source=chatgpt.com. [cit. 2025-01-03].
- [6] *Databázové systémy*. Online. Dkubinsky. Dostupné z: <https://www.dkubinsky.sk/clanok/uvoddostudia7>. [cit. 2025-01-03].
- [7] POKORNÝ, Jaroslav a Michal VALENTA. *Databázové systémy*. 2. přepracované vydání. Praha: Česká technika - nakladatelství ČVUT, 2020. ISBN 978-80-01-06696-6.
- [8] *Krátky úvod do SQL*. Online. Skliimea. 2021. Dostupné z: https://skillmea.sk/blog/kratky-uvod-do-sql?utm_source=chatgpt.com. [cit. 2025-01-03].
- [9] *Architektúra databázových systémov*. Online. Encyclopaedia Beliana. 1999. Dostupné z: https://beliana.sav.sk/heslo/architektura-databazovych-systemov?utm_source=chatgpt.com. [cit. 2025-01-03].
- [10] DATE, Chris. *Database Design and Relational Theory: Normal Forms and All That Jazz*. Second edition. Healdsburg, California, U.S.A: APress, 2019. ISBN 978-1-4842-5540-7.
- [11] DAVIDSON, Louis; MOSS, Jessica. *Pro SQL Server Relational Database Design and Implementation*. Apress, 2016. ISBN 9781484219720.
- [12] LAURENČÍK, Marek. *SQL: podrobný průvodce uživatele*. Praha: Grada Publishing. Průvodce (Grada), 2018. ISBN 978-802-7107-742.

- [13] *Inteligentná domácnosť od Loxone*. Online. Loxone. 1999. Dostupné z: <https://www.loxone.com/sksk/inteligentny-dom/>. [cit. 2025-01-03].
- [14] *Live Smart with Aqara*. Online. Aqara. 1999. Dostupné z: <https://www.aqara.com/en/>. [cit. 2025-01-03].
- [15] *What is Database Security?* Online. Intellipaat.com/. 2024. Dostupné z: <https://intellipaat.com/blog/database-security/>. [cit. 2025-01-03].

ZOZNAM OBRÁZKOV

Obrázek 1 Prvotný návrh ERD	47
Obrázek 2 Relačná schéma databázy	48
Obrázek 3 Zoznam vytvorených tabuliek v databáze	52
Obrázek 4 Tabuľka Zariadenie	53
Obrázek 5 Tabuľka Typ_zariadenia	53
Obrázek 6 Tabuľka Miestnosť	54
Obrázek 7 Tabuľka Scéna	54
Obrázek 8 Tabuľka Zariadenie_Scena	55
Obrázek 9 Tabuľka Akcia	55
Obrázek 10 Tabuľka Typ_akcie	56
Obrázek 10 Tabuľka Zariadenie_Akcia	57
Obrázek 12 Tabuľka Bezpečnostné_hlasenie	57
Obrázek 13 Tabuľka Typ_udalosti	58
Obrázek 14 Tabuľka Zariadenie_Hlasenie	59
Obrázek 15 Tabuľka Používateľ	59
Obrázek 16 Tabuľka Rola	60
Obrázek 17 Tabuľka Oprávnenie	60
Obrázek 18 Tabuľka Oprávnenie_role	61
Obrázek 19 Tabuľka Senzor	62
Obrázek 20 Tabuľka Typ_senzora	62
Obrázek 21 Tabuľka Jednotka	63
Obrázek 22 Tabuľka Senzor_hodnota	63
Obrázek 23 Tabuľka Zariadenie_senzor	64
Obrázek 24 Centrál - Loxone Config	68
Obrázek 25 Kúpeľňa - Loxone Config	71
Obrázek 26 Obývací izba - Loxone Config	73
Obrázek 27 Spálňa - Loxone Config	75
Obrázek 28 Okolie domu - Loxone Config	76
Obrázek 29 Úloha 1 - Loxone Config	78
Obrázek 30 Úloha 2 - Loxone Config	79
Obrázek 31 Úloha 3 - Loxone Config	80
Obrázek 32 Úloha 4 - Loxone Config	81
Obrázek 33 Úloha 5 – SSMS	81
Obrázek 34 Úloha 6 – SSMS	82

Obrázek 35 Úloha 7 – SSMS	82
Obrázek 36 Úloha 8 – SSMS	83

ZOZNAM TABULIEK

Tabulka 1 Loxone vs Aqara.....	38
--------------------------------	----

