



**ČVUT**

ČESKÉ VYSOKÉ  
UČENÍ TECHNICKÉ  
V PRAZE

**F3**

**Fakulta elektrotechnická**

**Bakalářská práce**

# **Základ aplikace pro skenování QR kódů**

**Anna Kachmasheva**

**Softwarové inženýrství a technologie**

**Leden 2024**

<https://github.com/AnnaKachmasheva/bp-app>

## Poděkování / Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

V Praze dne 10.01. 2024

.....

## Abstrakt / Abstract

The semester project deals with the issue of using QR codes in web applications for stock management in warehouses. The work is divided into chapters Introduction to QR codes, JavaScript libraries for scanning QR codes and Design of a web application. The chapter Introduction to QR codes describes basic information about QR codes, the history of their appearance, a description of the structure, types and their use in warehouses. The JavaScript library for scanning QR codes chapter describes and analyzes several popular libraries for scanning codes with a webcam. The Web Application Design chapter generally describes the analysis of the warehouse goods management system and includes a brief description of the implementation.

**Klíčová slova:** styl dokumentu; šablona; bakalářská, diplomová, disertační závěrečná práce; T<sub>E</sub>X.

Semestrální projekt se zabývá problematikou využití QR kódů ve webových aplikacích pro řízení zásob ve skladech. Práce je rozdělena na kapitoly Úvod do QR kódů, JavaScriptové knihovny pro skenování QR kódů a Návrh webové aplikace. Kapitola Úvod do QR kódů popisuje základní informace o QR kódech, historii jejich vzhledu, popis struktury, typů a jejich použití ve skladech. Kapitola JavaScriptové knihovny pro skenování QR kódů provádí popis a analýzu několika oblíbených knihoven pro skenování kódů pomocí webové kamery. Kapitola Návrh webové aplikace obecně popisuje analýzu systému řízení zboží ve skladu a obsahuje stručný popis implementace.

**Keywords:** js knihovny pro skenování QR kódů, skenování QR kódů použitím webové kamery, vyhledávání produktů pomocí QR kódu

**Title translation:** CTUstyle – the user manual (the OpT<sub>E</sub>X template for theses at CTU)

## / Obsah

<b>1 Úvod</b>	<b>1</b>
<b>2 Úvod do QR kódů</b>	<b>2</b>
2.1 Definice QR kódů . . . . .	2
2.2 Historie QR kódů . . . . .	2
2.3 Struktura QR kódů . . . . .	2
2.4 Typy QR kódů . . . . .	3
2.5 Používání QR kódů ve skladech . . . . .	4
<b>3 JavaScriptové knihovny pro skenování QR kódů</b>	<b>5</b>
3.1 jsqr . . . . .	5
3.2 @zxing/library . . . . .	6
3.3 react-qr-reader . . . . .	6
3.4 html5-qrcode . . . . .	6
3.5 qr-scanner . . . . .	7
3.6 Porovnání knihoven . . . . .	7
<b>4 Návrh webové aplikace</b>	<b>9</b>
4.1 Specifikace požadavků na systém evidence skladových zásob . . . . .	9
4.1.1 Funkční požadavky . . . . .	9
4.1.2 Nefunkční požadavky . . . . .	9
4.2 Uživatelé . . . . .	9
4.3 Analytický Datový Model . . . . .	10
4.4 Hlavní procesy . . . . .	11
4.5 Vývoj aplikace . . . . .	11

## Tabulky / Obrázky

<b>3.1</b> Výsledky skenování QR kódů v různých prohlížečích .....	7	<b>2.1</b> Structura QR kódu .....	3
<b>3.2</b> Průměrná doba skenování kódu .....	8	<b>2.2</b> Typy QR kódu .....	3
<b>3.3</b> Výsledky skenování poškoze- ných QR kódů .....	8	<b>3.1</b> UC .....	5
<b>3.4</b> Výsledky skenování kódů při špatném osvětlení .....	8	<b>4.1</b> UC .....	10
		<b>4.2</b> ADM .....	10
		<b>4.3</b> BPM .....	11

# Kapitola 1

## Úvod

V dnešní digitálně propojené době hrají čárové a QR kódy klíčovou roli v mnoha oblastech - od maloobchodu až po logistiku. Cílem tohoto semestrálního projektu je prozkoumat a využít potenciál těchto technologií prostřednictvím vývoje webové aplikace. Tato aplikace bude schopna nejen skenovat QR kódy, ale také spravovat databázi položek s nimi spojených.

Projekt se primárně zaměřuje na studium možností skenování QR kódů prostřednictvím klientských webových API. Bude nutné prozkoumat existující knihovny, posoudit jejich kompatibilitu a schopnosti, a zjistit, jak lze tyto technologie nejlépe implementovat do aplikace.

Hlavním výstupem bude návrh a implementace klientské webové aplikace. Tato aplikace by měla umožnit snímání QR kódů a správu s nimi spojených dat. Zároveň by měla být připravena na budoucí rozšíření o backend, který bude s klientem komunikovat pomocí REST API. Aplikace bude otestována v různých prohlížečích a nad různými typy kódů.

Dalším klíčovým aspektem projektu je uživatelské rozhraní. Aplikace by měla být intuitivní a snadno použitelná, aby vyhovovala technicky zdatným i méně zkušeným uživatelům. Design by měl být čistý, minimalistický a měl by usnadnit používání systému.

# Kapitola 2

## Úvod do QR kódů

QR kódy, neboli Quick Response kódy, jsou druhem dvourozměrného čárového kódu vyvinutého japonskou společností Denso Wave v roce 1994. Od té doby se staly důležitým prvkem v mnoha oblastech lidského života. Tato kapitola se zaměřuje na základní definici QR kódů, jejich historii, strukturu a celkový význam.

### 2.1 Definice QR kódů

QR kód je dvourozměrný typ čárového kódu, který lze snadno přečíst digitálním zařízením a ukládá informace jako řadu pixelů ve čtvercové mřížce, která se jeví jako černobílý vzor. QR kód představuje dvourozměrný maticový kód, což znamená, že informace jsou kódovány do sekvence černých a bílých čtverců – matice. Jeho dvourozměrnost spočívá v tom, že informace lze číst ve dvou směrech: vertikálně a horizontálně.

Existují dvě varianty QR kódů: statický, který nemění informace, a dynamický, který umožňuje úpravu obsahu bez zásahu do samotného QR kódu.

### 2.2 Historie QR kódů

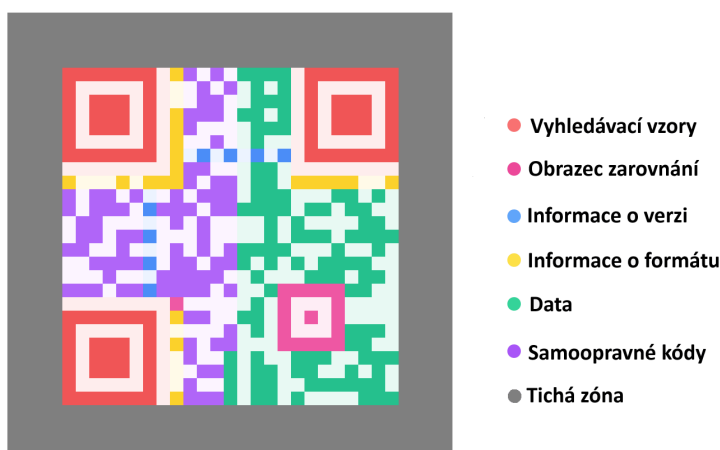
QR kódy byly původně používány ke sledování náhradních dílů ve výrobě automobilů a jejich součástí v japonském podniku Denso Wave, který je jednou z dceřiných společností Toyota. Zaměstnanci společnosti se potýkali s problémem pomalého čtení běžných čárových kódů, které sloužily k označení součástí, a jejich omezené kapacity pro uchování informací. Masahiro Hara začal pracovat na řešení těchto problémů v roce 1992. Chtěl vytvořit kód, který by byl viditelný, čitelný z jakéhokoli úhlu a uchovával více informací.

Masahiro Hara, inspirovaný svou láskou k deskové hře Go, zvolil podobu QR kódu, jak jej známe dnes. K vývoji této technologie použil open source přístup, díky kterému se QR kódy v 90. letech staly široce známými mimo japonský automobilový průmysl. V Japonsku byly použity na reklamních plakátech, billboardech, v časopisech, novinách a dokonce jako experimentální platební prostředek.

Teprve na počátku 20. století, s příchodem speciálních skenerů a aplikací pro mobilní telefony vybavených kamerami pro čtení QR kódů, se však tato technologie začala aktivně šířit do celého světa.

### 2.3 Struktura QR kódů

Vzor QR kódu ukládá zašifrovanou sekvenci dat v binárním formátu ve formě matice. Každé jednotlivé buňce mřížky je přiřazena hodnota podle barvy. Buňky jsou pak seskupeny do větších vzorů. Kódované datové klíče obsahují duplikáty, takže pokud je povrch QR kódu do určité míry poškozen, lze jej přečíst.



Obrázek 2.1. Struktura QR kódu.

Vyhledávací vzory jsou tři velké čtverce umístěné v rozích. Tyto vzory slouží jako identifikátor pro skener, signalizují mu přítomnost QR kódu a určují jeho pozici. Po detekci vzorů skener automaticky přechází do režimu čtení a začne dekódovat obsah kódu.

Značka zarovnání je další čtverec umístěný uvnitř QR kódu, avšak menší než vyhledávací vzory. Jeho účelem je indikovat směr čtení kódu skenerem.

Skener následně získá informace o verzi QR kódu. Existuje přibližně 40 verzí, přičemž nejmenší (verze 1) má rozměry 21x21 pixelů a největší (verze 40) 177x177 pixelů.

Důležitým krokem je analýza oblasti kolem vyhledávacích vzorů, kde se nachází informace o formátu kódování (písmena, čísla nebo Kanji) a kódy pro opravu chyb. Tyto kódy umožňují skeneru korigovat a správně interpretovat informace i při poškození QR kódu o 30 procent.

Synchronizační řádky obsahují informace o velikosti matice a umožňují číst kód i na nerovném povrchu, zajišťují synchronizaci při čtení.

Datové buňky, tvořené černobílými moduly, obsahují zakódované informace, například text, telefonní číslo nebo odkazy. Počet modulů přímo koresponduje s délkou zakódovaných znaků. Kód dokáže zašifrovat tři kilobajty binárního kódu nebo asi čtyři stránky textových informací – více než sedm tisíc čísel nebo čtyři tisíce písmen.

Bílé pole kolem kódu a uvnitř vyhledávacích vzorů představuje odsazení, které je klíčové pro správnou interpretaci mezi kódem a informacemi v buňkách. Bez tohoto odsazení by skener nemohl efektivně rozlišit mezi strukturou kódu a obsahem buňek.

## 2.4 Typy QR kódů

QR kódy se vyznačují různými formáty a strukturami, každý s unikátními vlastnostmi a využitím. Níže je uveden popis několika typů QR kódů.



Obrázek 2.2. Typy QR kódu.



Model 2 je vylepšením Modelu 1. Díky vestavěnému vzoru zarovnání lze tento typ kódu číst i na zakřivených površích nebo pod zkreslenými úhly. Tento kód může kódovat až 7089 číslic, maximální verze je 40 (177 x 177 modulů).

FrameQR umožňuje vložení obrázků, log, nebo jiných grafických prvků do rámu kolem standardního QR kódu.

SQRC vypadá jako kód Modelu 2, obsahuje však veřejná a soukromá data. Soukromá data lze číst pouze pomocí speciální čtečky, která má kryptografický klíč zajišťující ochranu dat.



Použití kódů ve skladu především zjednodušuje příjem nového zboží. Označení každé položky zboží, skupiny zboží nebo celé šarže umožňuje zkrátit dobu příjmu zboží a minizuje počet chyb.

Tradiční metody inventarizace jsou pracné a náchylné k chybám. Použití QR kódů umožňuje provádět inventuru rychle a s minimem chyb, což výrazně zvyšuje přesnost účetnictví.

QR kódy se navíc snadno přizpůsobují měnícím se obchodním potřebám a umožňují rychlé změny v systémech řízení zásob.

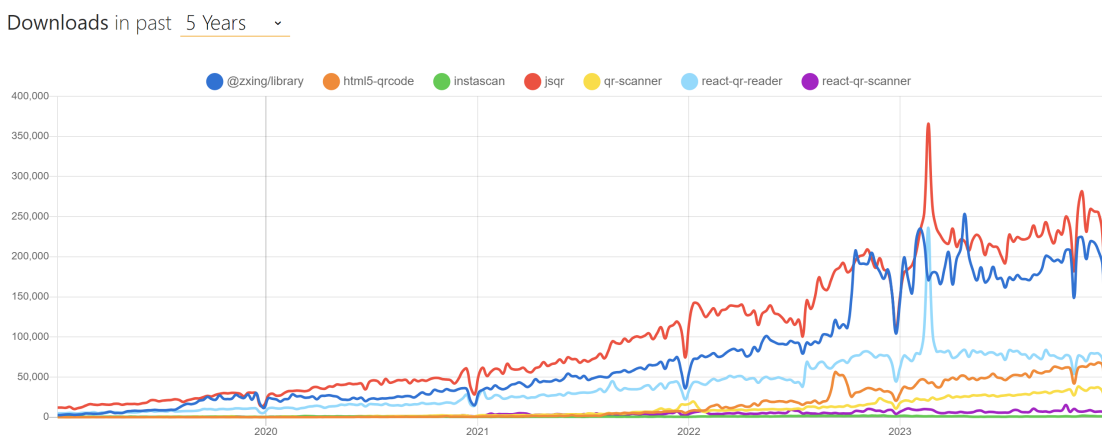
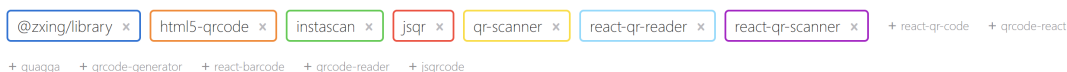
Implementace QR kódů ve skladech je krokem k vytvoření chytřejšího, automatizovanějšího a efektivnějšího řízení zásob, které je klíčovým faktorem úspěchu v dnešním dynamickém obchodním prostředí. Tato technologie nejen usnadňuje každodenní skladové operace, ale také otevírá nové příležitosti pro optimalizaci a inovace v logistice a řízení dodavatelského řetězce.

# Kapitola 3

## JavaScriptové knihovny pro skenování QR kódů

Tato kapitola se zaměřuje na představení a analýzu několika JavaScriptových knihoven určených pro skenování QR kódů. Budou prozkoumány jejich hlavní charakteristiky, funkčnost, výhody a možná omezení. Dále budou uvedeny příklady implementace těchto knihoven ve webové aplikaci pro správu zboží ve skladu. Tyto příklady budou testovány a vzájemně porovnávány.

Byly analyzovány trendy ve využívání knihoven. Jsou znázorněny na obrázku 3.1. Na základě těchto statistik bylo vybráno 5 nejčastěji používaných knihoven.



Obrázek 3.1. Populární knihovny pro skenování QR kódů

### 3.1 jsqr

Knihovna jsQR je čistě JavaScriptová open source knihovna pro čtení QR kódu navržená tak, aby byla flexibilní ve více prostředích. Dokáže zpracovat surové obrázky a je schopen najít, extrahovat a analyzovat jakékoli QR kódy, které se v nich nacházejí. Díky tomu je ideální volbou pro projekty, kde potřebujete skenovat QR kódy z různých zdrojů, jako je stream z webové kamery, obrázek nahraný uživatelem nebo dokonce uvnitř interního procesu Node.js. Knihovna je zcela samostatná a nezávisí na kódu specifickém pro platformu.

Při dekódování QR kódu může vrátit různá data, včetně nezpracovaných bajtů, řetězcových dat a informací o struktuře a obsahu QR kódu.

jsqr je aktivně udržována knihovna a příspěvky do projektu jsou vítány. Je licencována pod licencí Apache-2.0.



Knihovna momentálně již není v aktivním vývoji a neexistují žádné plány na budoucí změny. Stále se však jedná o spolehlivý nástroj pro čtení a generování kódů.

Instalace je jednoduchá, pomocí npm nebo yarn, a knihovna nabízí několik způsobů, jak integrovat skenování čárových kódů do aplikací, ať už v prohlížeči nebo v rámci Node.js. Je to všestranný nástroj vhodný pro různé aplikace včetně skenování a generování čárových kódů.



Knihovna byla testována a je kompatibilní s hlavními prohlížeči, jako je Chrome, Firefox a Safari. K dispozici jsou pokyny pro zajištění kompatibility knihovny se staršími prohlížeči.

Je to open-source projekt distribuovaný pod licencí MIT, který vítá příspěvky od komunity.



Knihovna je navržena tak, aby byla kompatibilní s více prohlížeči a zařízeními, takže je všestranná pro širokou škálu webových aplikací. Poskytuje jednoduché a přímočaré rozhraní API, které usnadňuje integraci do stávajících webových projektů s minimální konfigurací. Vývojáři mají flexibilitu přizpůsobit vzhled a chování skeneru tak, aby odpovídal designu jejich aplikace.

Celkově knihovna `html5-qrcode` nabízí pohodlné, efektivní a všestranné řešení pro integraci funkce skenování QR kódu do webových aplikací.

### 3.5 qr-scanner

Knihovna poskytuje robustní a efektivní způsob integrace funkce skenování QR kódu do webových aplikací. Využívá moderní webové technologie, které umožňují bezproblémové dekódování QR kódů z různých zdrojů.

Je optimalizována tak, aby efektivně zvládala různá rozlišení kamery a světelné podmínky. Dokáže automaticky vybrat nejvhodnější kameru na zařízení, což je užitečné zejména v zařízeních s více kamerami. Kromě skenování v reálném čase knihovna také podporuje skenování QR kódů z obrázků.

Implementuje optimalizované algoritmy pro rychlé a přesné dekódování QR kódu, snižuje latenci a zlepšuje uživatelský zážitek.

Nabízí přímočaré rozhraní API, které zjednodušuje proces integrace skenování QR kódu do webových projektů. Navržena tak, aby fungovala v celé řadě prohlížečů.

Knihovna QR-Scanner je ideálním řešením pro vývojáře, kteří chtějí do svých webových aplikací začlenit efektivní a spolehlivé skenování QR kódu se zaměřením na výkon a uživatelskou zkušenost.

### 3.6 Porovnání knihoven

V tomto projektu výše uvedené knihovny 3 použity ke skenování QR kódů produktů ve skladu. Proto se předpokládá, že QR kódy jsou stejného typu. Je zakódován řetězec obsahující parametry produktu. Očekává se, že řetězec může obsahovat písmena s diakritikou.

Během testování porovnání knihoven byl například použit řetězec: „id:202, price:19.99, minQuantity:20, quantity:200, photo:mockingbird\_audiobook.jpg, format: Audio-book, author:HarperLee, format:Paperback, genre:Historická beletrie“.

Níže je srovnání knihoven na základě parametrů, jako je práce v různých prohlížečích, pravděpodobnost detekce kódu, rychlost detekce kódu, testování za špatných světelných podmínek a práce s kódy různých velikostí.

Přesnost rozpoznávání kódu je klíčovým faktorem při skenování kódů pomocí webové kamery. Srovnávací analýza se zaměřuje na to, jak úspěšně jsou QR kódy rozpoznávány v různých prohlížečích. Podle statistik získaných ze zdroje 1 byly vybrány 4 nejoblíbenější prohlížeče. Pro každý prohlížeč bylo provedeno 30 skenů kódu.

knihovna	Chrome	Safari	Edge	Firefox
jsqr	100 %	100 %	100 %	100 %
@zxing/library	100 %	100 %	100 %	100 %
react-qr-reader	100 %	100 %	100 %	100 %
html5-qrcode	100 %	100 %	100 %	100 %
qr-scanner	100 %	100 %	100 %	100 %

**Tabulka 3.1.** Výsledky skenování QR kódů v různých prohlížečích z [??].

Dalším důležitým parametrem je rychlost rozpoznání kódu. V prohlížeči Chrome bylo naskenováno 50 různých kódů. Pro každý kód byl zaznamenán čas skenování. Výsledky jsou uvedeny v tabulce níže.

Vzhledem k tomu, že skenování kódů bude prováděno ve skladech, je velká pravděpodobnost, že dojde k poškození kódů např. při přesunu zboží. Proto dalším důležitým parametrem při porovnávání knihoven je, jak dobře jsou schopny pracovat s takovými

knihovna	průměrná doba v sekundách
jsqr	1
@zxing/library	méně než 1
react-qr-reader	1
html5-qrcode	2
qr-scanner	méně než 1

**Tabulka 3.2.** Průměrná doba skenování kódu z [??].

knihovna	úspěšnost
jsqr	99%
@zxing/library	100%
react-qr-reader	98%
html5-qrcode	90%
qr-scanner	99%

**Tabulka 3.3.** Výsledky skenování poškozených QR kódů z [??].

QR kódy. V prohlížeči Chrome bylo naskenováno 50 různých kódů s různými typy deformací. Výsledky jsou v tabulce níže.

V reálném prostředí skladu je možné, že skenování kódu bude prováděno při slabém osvětlení. Proto dalším důležitým kritériem pro porovnávání Knihoven je, jak dobře se vypořádají s rozpoznáváním QR kódů za špatných světelných podmínek.

knihovna	úspěšnost
jsqr	100 %
@zxing/library	100 %
react-qr-reader	100 %
html5-qrcode	100 %
qr-scanner	100 %

**Tabulka 3.4.** Výsledky skenování kódů při špatném osvětlení z [??].

Srovnání knihoven ukázalo, že mezi nimi existuje jen minimální rozdíl. Všechny tyto knihovny podporují většinu prohlížečů a pracují efektivně i při nedostatečném osvětlení. Při skenování kódu v prohlížeči Chrome vykazovaly knihovny @zxing/library a qr-scanner nejlepší výsledky z hlediska doby skenování. Naopak knihovna react-qr-reader dosáhla mírně horších výsledků ve srovnání s ostatními, zejména při práci s poškozenými kódy.

Během provedených experimentů byly odhaleny určité rozdíly mezi těmito knihovnami. Například knihovna html5-qrcode má problémy se skenováním šikmých kódů a vyžaduje, aby byl QR kód přiblížen kameře, aby byl úspěšně detekován.

U knihovny qr-scanner trvá několik sekund, než otevře stream videa, což způsobuje, že po otevření modálního okna nejsou po několika sekundách zobrazeny žádné informace. Tato knihovna schopna detekovat kódy i na větší vzdálenost od kamery.

Výsledky získané v této podkapitole jsou přibližné, protože byly získány pomocí malého počtu pokusů o naskenování QR kódu (30 pro každý test).

# Kapitola 4

## Návrh webové aplikace

Webové aplikace pro skenování QR kódu, které běží přímo v prohlížeči, poskytují jedinečné výhody oproti jiným metodám skenování, jako jsou mobilní aplikace nebo vyhrazené skenery. Jsou dostupné na jakémkoli zařízení s webovým prohlížečem, což z nich činí univerzální řešení pro uživatele s různými typy zařízení, včetně telefonů, tabletů a stolních počítačů.

Kromě toho používání webových aplikací snižuje potřebu fyzických skenerů a dalšího vybavení, což pomáhá snížit elektronický odpad a celkové náklady.

### 4.1 Specifikace požadavků na systém evidence skladových zásob

Webová aplikace vyvinutá v této práci musí uživatelům poskytovat určité funkce. Důraz je kladen na řízení zásob pomocí QR kódů.

#### 4.1.1 Funkční požadavky

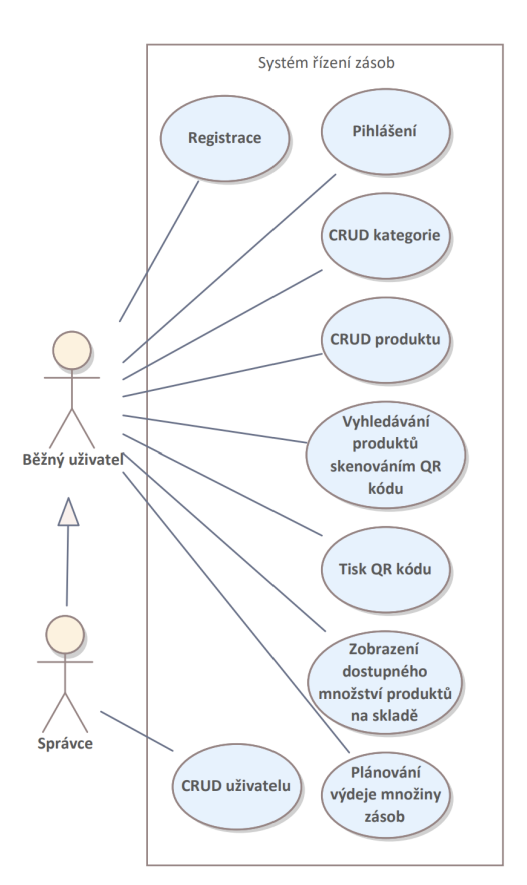
- Systém bude schopen naskenovat QR kód a tento kód použít k nalezení produktu.
- Systém bude umožňovat CRUD operace do tabulky skladových zásob.
- Systém bude umožňovat příjem, výdej zboží a kontrolu úplnosti výdeje celé podmnožiny.
- Systém bude označovat podmnožiny existujících zásob.
- Systém bude umožňovat plánování výdeje množiny zásob.
- Součástí aplikace bude jednoduchá autorizace.

#### 4.1.2 Nefunkční požadavky

- Systém bude implementován jako webová aplikace.
- K ukládání dat bude použita relační databáze.
- Pro komunikaci mezi serverem a klientem bude použito REST API.

### 4.2 Uživatelé

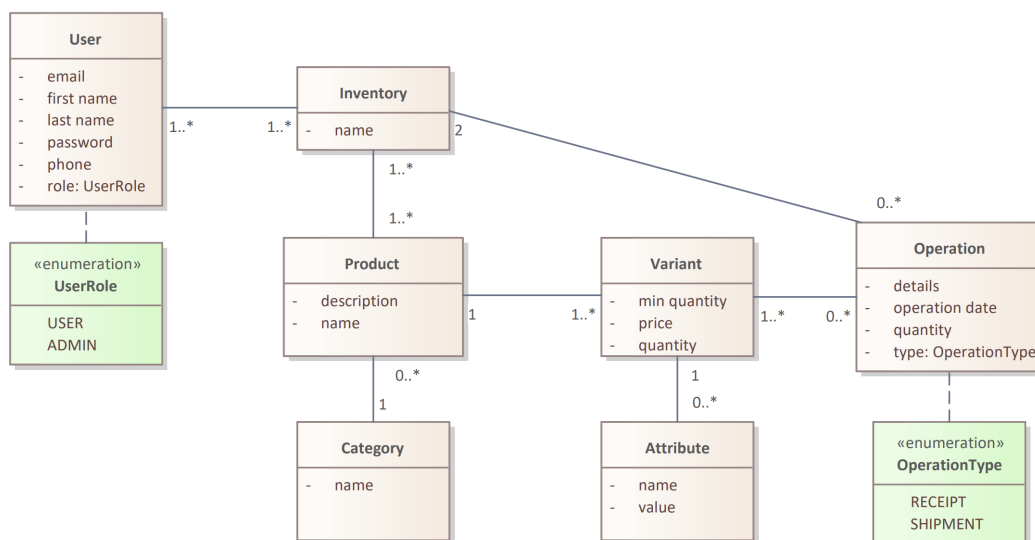
V systému bude mít autorizovaný uživatel 2 hlavní role. Jedná se o roli běžného uživatele a roli správce. Uživatel bude kontrolovat zboží ve skladu, proto mezi jeho hlavní pracovní funkce patří příjem zboží a jeho výdej, práce s tabulkou všech zásob. Kromě možností, které bude mít běžný uživatel, správce bude mít možnost spravovat další uživatele. Diagram UC je uveden na obrázku 4.1.



Obrázek 4.1. Use Case Diagram.

### 4.3 Analytický Datový Model

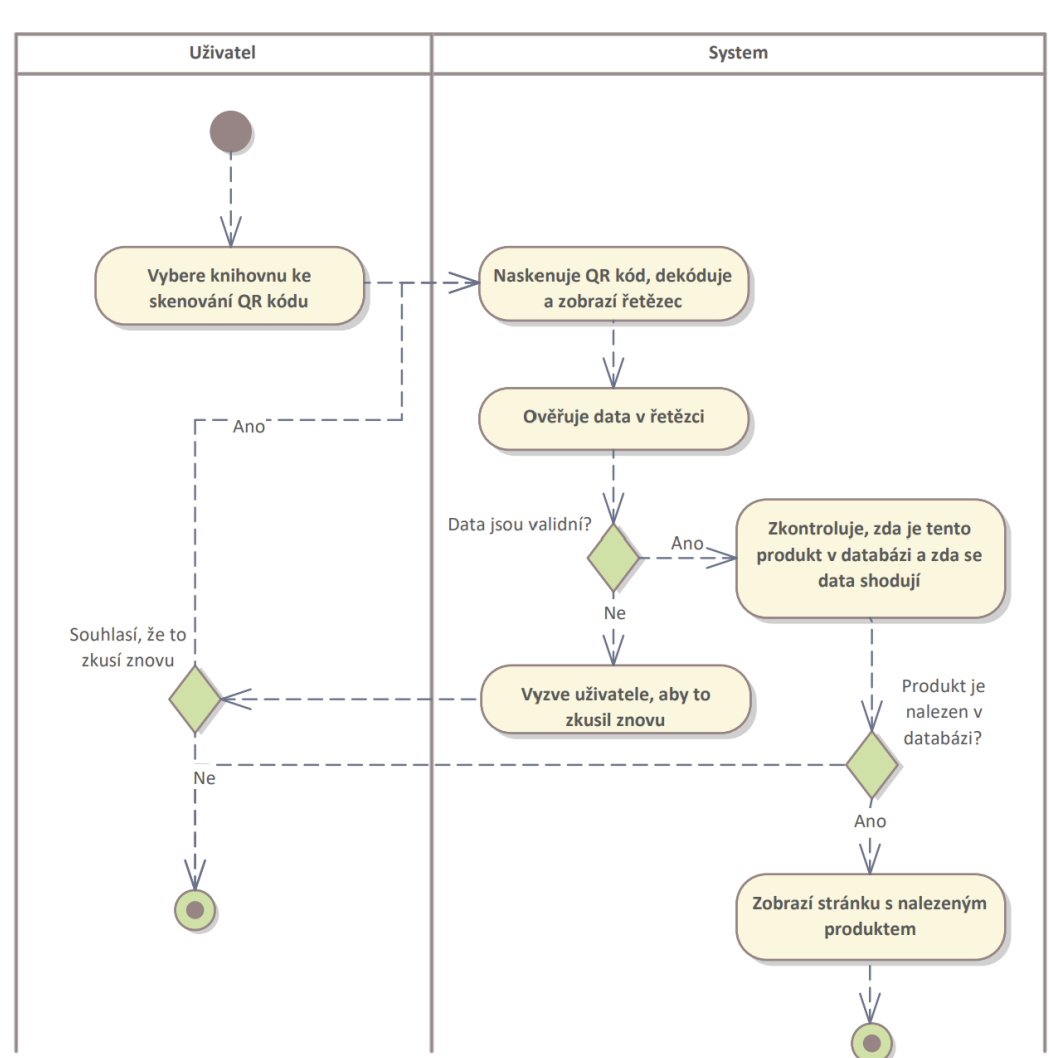
Společně s UC diagramem byl vytvořen tento model, který je uveden na obrázku 4.2. Vzhledem k tomu, že jedním z požadavků na systém je použití relační databáze, bude v budoucnu sloužit k návrhu datového modelu.



Obrázek 4.2. ADM diagram.

## 4.4 Hlavní procesy

Hlavním procesem v rámci systému bude naskenování kódu, získání informací z něj a následné vyhledání produktu v databázi. Popsaný proces je znázorněn na obrázku 4.3.



**Obrázek 4.3.** BPM diagram.

Procesy spojené s příjmem a výdejem zásob a plánováním těchto operací mají také význam. V rámci tohoto semestrálního projektu však tyto procedury nebyly implementovány; jejich realizace proběhne v následujícím semestru.

## 4.5 Vývoj aplikace

Tato podkapitola je věnována implementaci systému pro evidence skladových zásob. Při návrhu webové aplikace jsem vycházela z diagramů UC a ADM. K vývoji front-endové části jsem použila framework React.