# SRS LexLector

## Software Requirements Specification

### 1. Úvod

Tento dokument specifikuje požadavky na webovou aplikaci LexLector určenou pro čtení a jazykové učení prostřednictvím textových dokumentů. Aplikace umožní uživatelům nahrávat a číst soubory ve formátech EPUB, DOCX a TXT. Klíčovou funkcí je překlad neznámých slov a vět v reálném čase s možností využití vlastního slovníku i externího API. Dokument reflektuje současné požadavky a je otevřený budoucímu rozšíření systému.

**1.1 Účel dokumentu**

Cílem tohoto dokumentu je podrobně popsat funkční a nefunkční požadavky na vývoj webové aplikace sloužící jako interaktivní čtečka cizojazyčných textů. Dokument bude sloužit jako referenční materiál pro vývojový tým, testery a projektového manažera během celého životního cyklu softwaru.

**1.2 Cílové publikum**

Dokument je určen pro členy vývojového týmu (frontend, backend), testery, dokumentaristy, analytiky a další zúčastněné strany, které se podílejí na návrhu, vývoji a údržbě systému. Rovněž může sloužit jako podklad pro budoucí rozšiřování systému.

**1.3 Rozsah systému**

Navrhovaná aplikace poskytne uživatelsky přívětivou platformu pro čtení elektronických knih a textových souborů s důrazem na podporu jazykového vzdělávání. Uživatel bude moci kliknutím na slovo nebo větu získat překlad, který je primárně načten z lokální databáze. V případě, že překlad není dostupný, aplikace využije ChatGPT API k získání překladu, který se následně uloží do lokální databáze pro budoucí použití. Uživatelé si budou moci tvořit vlastní knihovnu a vlastní slovník. Aplikace bude dostupná i bez registrace, přičemž registrovaní uživatelé získají přístup k pokročilejším funkcím, jako je synchronizovaná knihovna, osobní nastavení a správa slovníku.

**1.4 Definice, zkratky a zkratková slova**

* **EPUB** – Electronic Publication, formát pro elektronické knihy
* **DOCX** – formát dokumentů Microsoft Word
* **TXT** – prostý textový soubor
* **API** – Application Programming Interface
* **DB** – Databáze
* **UI** – Uživatelské rozhraní
* **ChatGPT** – model umělé inteligence poskytující jazykové odpovědi a překlady

**1.5 Přehled dokumentu**

Tento dokument dále obsahuje podrobný popis funkčních a nefunkčních požadavků (kapitola 2 a 3), návrh uživatelských scénářů a případů užití (kapitola 4), popis rozhraní systému (kapitola 5), a specifikaci dalších technických a provozních požadavků.

### 2. Funkční požadavky

Funkční požadavky jsou rozděleny podle hlavních modulů systému LexLector:

**2.1 Modul Čtečka**

* Uživatel může nahrát soubor ve formátu EPUB, DOCX nebo TXT.
* Aplikace zobrazí obsah souboru ve čtečce optimalizované pro čtení.
* Po kliknutí na slovo se zobrazí překlad v bublině.
* Pokud překlad existuje v databázi, zobrazí se uložený překlad.
* Pokud překlad neexistuje, odešle se požadavek na ChatGPT API, výsledek se zobrazí a zároveň uloží.
* Možnost zobrazení překladu celé věty.

**2.2 Modul Knihovna**

* Registrovaný uživatel má svou osobní knihovnu s nahranými knihami.
* Možnost přidávat a mazat knihy.
* Možnost přidělovat knihám tagy a třídit je podle různých kritérií.
* Aplikace si pamatuje pozici, kde uživatel přestal číst.

**2.3 Modul Slovník**

* Uživatel má přístup ke svému osobnímu slovníku.
* Slovník obsahuje slova, která uživatel překládal.
* Možnost ručně přidat nebo upravit překlad slova.
* Podpora specifických překladů pro jednotlivé knihy nebo kontexty (např. fantasy termíny).

**2.4 Modul Uživatelský účet**

* Možnost registrace a přihlášení uživatele.
* Uživatel má přístup ke své knihovně, slovníku a nastavení z různých zařízení.
* Možnost upravit osobní údaje a jazykové preference.
* Podpora vícejazyčného rozhraní aplikace.

**2.5 Modul Překladový engine**

* Překlad probíhá ve dvou krocích: hledání v DB a případně dotaz na API.
* Výsledky překladu se ukládají do databáze.
* Při neúspěchu požadavku na API se zobrazí chybová hláška.

**2.6 Modul Rozhraní**

* Aplikace má vícejazyčné uživatelské rozhraní.
* Podpora lokalizace pro alespoň angličtinu a češtinu.
* Responzivní design přizpůsobený pro čtení na desktopu i mobilních zařízeních.

### 3. Nefunkční požadavky

**3.1 Výkonnost**

* Aplikace musí být schopna zpracovat a zobrazit běžné knihy (do 10 MB) během několika sekund.
* Požadavek na překlad nesmí trvat déle než 5 sekund (při dostupnosti API).
* Odezva UI při běžné interakci (klikání, překlady, přechod mezi stránkami) by neměla překročit 500 ms.

**3.2 Škálovatelnost**

* Backend musí být navržen s ohledem na budoucí rozšíření (více uživatelů, více jazyků, více překladových API).
* Možnost rozdělit architekturu na mikroservisy v budoucnu.

**3.3 Dostupnost a spolehlivost**

* Aplikace by měla být dostupná minimálně 99 % času.
* V případě výpadku překladu přes API bude uživatel informován a překlad nebude proveden.

**3.4 Bezpečnost**

* Veškerá komunikace musí být šifrována (HTTPS).
* Uživatelé musí mít bezpečně uložená hesla (např. pomocí bcrypt).
* API klíče (např. pro ChatGPT) nesmí být přístupné z klientské části.

**3.5 Přenositelnost a kompatibilita**

* Aplikace bude fungovat v moderních prohlížečích (Chrome, Firefox, Edge, Safari).
* Aplikace bude fungovat na desktopových i mobilních zařízeních.

**3.6 Lokalizace a internacionalizace**

* Veškeré texty v UI budou spravovány pomocí jazykových balíčků.
* Aplikace bude připravena na přidání dalších jazyků bez zásahu do kódu.

**3.7 Údržba a rozšiřitelnost**

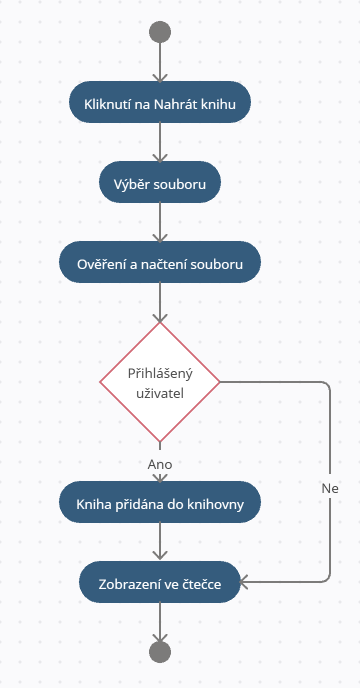
* Kód bude strukturován dle MVC/MVVM principů.
* Použití verzovacího systému (např. Git).
* Dokumentace k API a modulům bude součástí projektu.

### 4. Uživatelské scénáře a případy užití

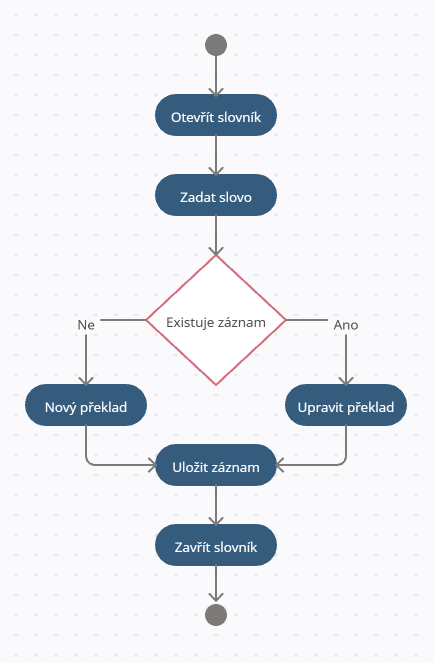
**Obsah obrázku text, snímek obrazovky, diagram, řada/pruh

Obsah vygenerovaný umělou inteligencí může být nesprávný.Use Case 4.1: Přeložení slova při čtení**

* **Název:** Přeložení slova
* **Primární aktér:** Uživatel
* **Cíl:** Získat překlad neznámého slova kliknutím na něj
* **Předpoklady:** Uživateli se zobrazuje kniha ve čtečce
* **Popis:** Uživatel klikne na slovo, aplikace zjistí, zda je překlad v databázi. Pokud ne, požádá ChatGPT API o překlad a výsledek uloží.
* **Hlavní scénář:**
  1. Uživatel klikne na slovo
  2. Systém zkontroluje lokální databázi překladů
  3. Pokud existuje, zobrazí překlad
  4. Pokud neexistuje, odešle požadavek na API
  5. Zobrazí překlad z API
  6. Uloží výsledek do databáze
* **Alternativní scénář:**
  1. 4a. Pokud API není dostupné → zobrazí chybovou zprávu

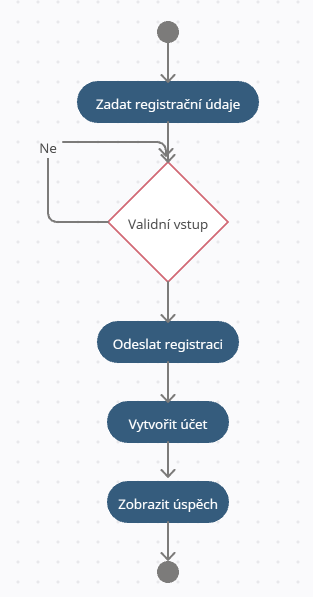
**Use Case 4.2: Zobrazit knihu**

* **Název:** Zobrazení knihy
* **Primární aktér:** Uživatel
* **Cíl:** Zobrazit knihu
* **Předpoklady:** Uživatel je přihlášen
* **Popis:** Uživatel nahraje podporovaný soubor a pokud je přihlášený, přidá ho do své knihovny
* **Hlavní scénář:**
  1. Uživatel klikne na „Nahrát knihu“
  2. Vybere soubor (EPUB, DOCX, TXT)
  3. Systém zvaliduje a načte soubor
  4. Pokud je uživatel přihlášen, je kniha přidána do uživatelovy knihovny
  5. Zobrazí se náhled ve čtečce
* **Alternativní scénář:**
  1. 3a. Soubor není podporován → zobrazí se chybová hláška

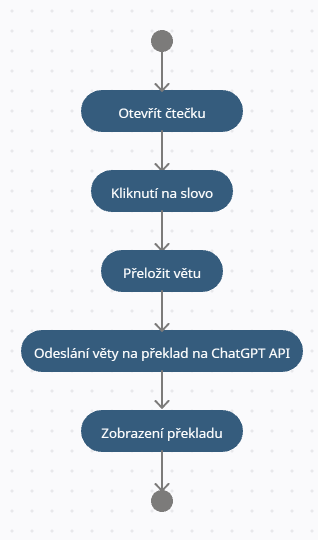


**Use Case 4.3: Správa osobního slovníku**

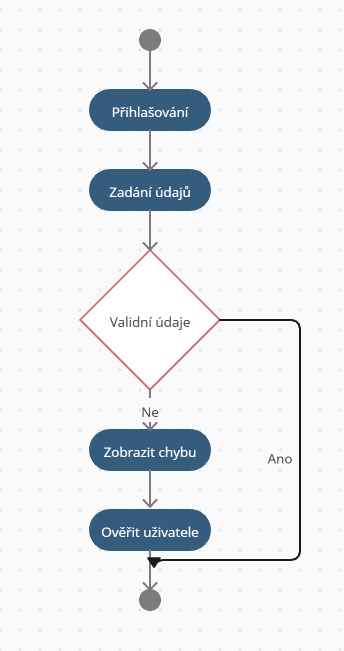
* **Název:** Úprava překladu slova
* **Primární aktér:** Registrovaný uživatel
* **Cíl:** Přidat nebo upravit vlastní překlad
* **Předpoklady:** Uživatel je přihlášen
* **Popis:** Uživatel si ručně přidá překlad slova nebo upraví existující
* **Hlavní scénář:**
  1. Uživatel otevře svůj slovník
  2. Klikne na „Přidat slovo“ nebo vybere existující
  3. Zadá slovo a jeho překlad
  4. Uloží změny
  5. Systém potvrdí uložení

**Use Case 4.4: Registrace nového uživatele**

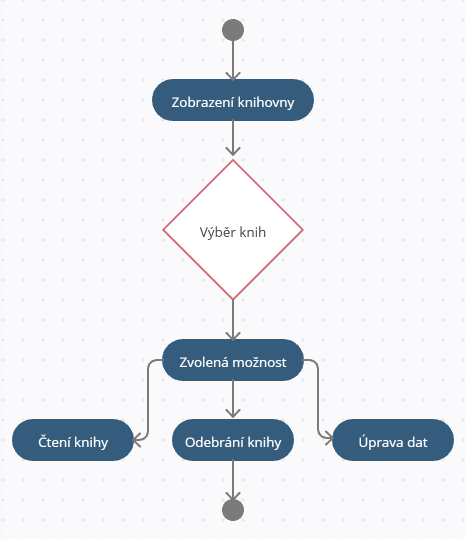
* **Název:** Registrace
* **Primární aktér:** Neregistrovaný uživatel
* **Cíl:** Vytvořit si účet v systému
* **Předpoklady:** Uživatelský účet ještě neexistuje
* **Hlavní scénář:**
  1. Uživatel přejde na stránku registrace
  2. Zadá e-mail a heslo
  3. Potvrdí souhlas s podmínkami
  4. Odešle formulář
  5. Systém vytvoří účet
  6. Přihlásí uživatele a přesměruje ho do aplikace
* **Alternativní scénář:**
  1. 2a. E-mail už existuje → zobrazí chybu

**Use Case 4.5: Překlad celé věty**

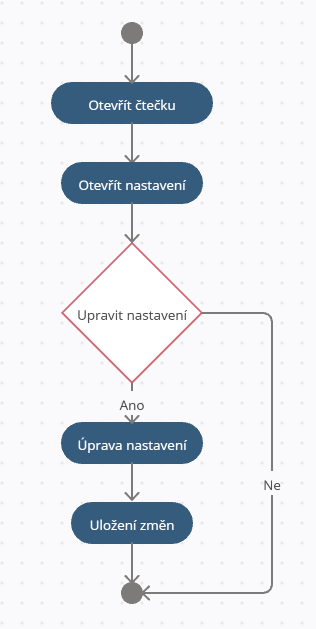
* **Název:** Překlad věty
* **Primární aktér:** Uživatel
* **Cíl:** Získat překlad celé věty, ve které se nachází neznámé slovo
* **Předpoklady:** Kniha je otevřená ve čtečce
* **Hlavní scénář:**
  1. Uživatel klikne na slovo
  2. Systém identifikuje větu obsahující dané slovo
  3. Uživatel zvolí možnost „Přeložit větu“
  4. Systém odešle celou větu do ChatGPT API
  5. Zobrazí se překlad věty v bublině pod textem
* **Alternativní scénář:**
  1. 4a. Pokud API není dostupné → zobrazí se chybová zpráva

**Use Case 4.6: Přihlášení uživatele**

* **Název:** Přihlášení
* **Primární aktér:** Registrovaný uživatel
* **Cíl:** Získat přístup k osobní knihovně a nastavením
* **Předpoklady:** Uživatel má existující účet
* **Hlavní scénář:**
  1. Uživatel přejde na přihlašovací stránku
  2. Zadá e-mail a heslo
  3. Klikne na „Přihlásit se“
  4. Systém ověří přihlašovací údaje
  5. Uživatel je přihlášen a přesměrován do aplikace
* **Alternativní scénář:**
  1. 4a. Neplatné údaje → zobrazí se chybová hláška

**Use Case 4.7: Správa knihovny**

* **Název:** Správa knihovny
* **Primární aktér:** Registrovaný uživatel
* **Cíl:** Organizovat knihy – mazat, tagovat, řadit
* **Předpoklady:** Uživatel je přihlášen
* **Hlavní scénář:**
  1. Uživatel otevře knihovnu
  2. Vybere knihu nebo více knih
  3. Provádí akce: přidání tagu, změna názvu, smazání
  4. Systém aktualizuje knihovnu

**Use Case 4.8: Nastavení uživatelského prostředí**

* **Název:** Změna nastavení čtečky
* **Primární aktér:** Registrovaný uživatel
* **Cíl:** Přizpůsobit si vzhled a chování čtečky
* **Předpoklady:** Uživatel je přihlášen
* **Hlavní scénář:**
  1. Uživatel přejde do nastavení
  2. Změní např. velikost písma, rozestupy, jazyk rozhraní
  3. Uloží nastavení
  4. Změny se aplikují na čtečku
  5. Nastavení se uloží do profilu

### 5. Popis rozhraní systému

**5.1 Uživatelské rozhraní (UI)**

* Hlavní obrazovka čtečky s navigací mezi kapitolami a možností kliknutí na slova.
* Bublina s překladem se zobrazením slova a jeho významu.
* Knihovna uživatele s náhledy knih, možností filtrování, tagování a mazání.
* Slovník s tabulkovým výpisem přeložených slov a možností úprav.
* Nastavení aplikace: jazyk UI, velikost písma, režim (světlý/tmavý).

**5.2 Vnitřní rozhraní mezi moduly**

* Modul čtečky komunikuje s modulem překladu.
* Modul knihovny pracuje s databází souborů a metadat.
* Modul slovníku interaguje s databází uživatelských překladů.

**5.3 API rozhraní (plánováno)**

* Do budoucna bude možné vytvořit REST API pro interakci s aplikací.
* Předpokládané endpointy: /api/translate, /api/books, /api/dictionary.
* API bude vracet data ve formátu JSON.

### 6. Technické a provozní požadavky

**6.1 Hardwarové požadavky**

* Server s PHP 8+ a MySQL 8+.
* Minimálně 2 GB RAM a SSD úložiště.

**6.2 Softwarové požadavky**

* Webový server: Apache (preferovaný).
* PHP s podporou rozšíření pro práci s dokumenty (zip, xml, mbstring…)
* Databázový server: MySQL nebo MariaDB.

**6.3 Závislosti na třetích stranách**

* ChatGPT API (OpenAI) pro překlad slov a vět.
* PHP knihovny pro zpracování EPUB, DOCX (např. PHPWord, epub-reader).

**6.4 Zálohování a provozní režim**

* Denní automatické zálohy databáze a nahraných knih.
* Uložení souborů mimo veřejný web root.
* Aplikace bude dostupná 24/7 s výpadky pouze při údržbě.

**6.5 Monitorování a logování**

* Logování chyb, API požadavků a uživatelských akcí.
* Statistiky přístupů a překládání slov.
* Upozornění při překročení limitu API nebo selhání zálohování.