## Dokumentace specifikace požadavků

### 1.1 Účel dokumentu

Tento dokument specifikuje softwarové požadavky na systém **InternHub**. Jeho cílem je jednoznačně vymezit funkčnost, uživatelské role, rozhraní a technická omezení systému. Dokument slouží jako referenční rámec pro vývoj, testování i schvalování výsledného softwarového řešení.

### 1.2 Rozsah produktu

### InternHub je webová aplikace určená k podpoře a digitalizaci správy odborných praxí studentů. Systém slouží jako centralizovaná platforma, která propojuje studenty, organizace nabízející praxe a akademické pracovníky odpovědné za evidenci a schvalování těchto praxí.

### 1.3 Definice, zkratky a akronymy

* **UJEP** – Univerzita Jana Evangelisty Purkyně v Ústí nad Labem
* **STAG** – Studijní agenda univerzity
* **ARES** – Administrativní registr ekonomických subjektů
* **UI** – Uživatelské rozhraní
* **UX** – Uživatelský zážitek
* **API** – Aplikační programové rozhraní
* **CRUD** – Create, Read, Update, Delete
* **JWT** – JSON Web Token
* **UAT** – User Acceptance Testing
* **BDD** – Behavior Driven Development
* **ERD** – Entity-Relationship Diagram
* **UML** – Unified Modeling Language

### 1.4 Reference

* Jan Viola – Use Case Diagram *(odkaz bude doplněn)*
* Martin Vlnas – Prototyp uživatelského rozhraní ve Figma *(odkaz bude doplněn)*
* Dokumentace k API systémů STAG a ARES

### 1.5 Přehled dokumentu

Tento dokument je strukturován podle standardní šablony SRS dle IEEE. Obsahuje úvodní přehled, charakteristiku produktu, detailní funkční a nefunkční požadavky, popis rozhraní a přehled metod ověřování. Dokument slouží jako základní vodítko pro celý životní cyklus vývoje systému InternHub.

## 2. Přehled produktu

### 2.1 Perspektiva produktu

InternHub je samostatná webová aplikace navržená k digitalizaci a centralizované správě odborných praxí studentů. Aplikace je postavena na modulární architektuře s odděleným frontendem a backendem a je připravena k případnému rozšíření pro potřeby dalších kateder a fakult.

### 2.2 Funkce produktu

InternHub poskytuje funkcionality rozdělené dle uživatelských rolí:

* **Student:**
  + Prohlížení a filtrování nabídek praxí
  + Podání přihlášky na praxi
  + Sledování stavu přihlášek
  + Možnost odmítnout nebo zrušit přihlášku
* **Organizace (firma):**
  + Registrace subjektu pomocí IČO a validace přes ARES
  + Vytváření, úprava a mazání nabídek praxí
  + Sledování a správa přijatých přihlášek
* **Katedra / školní koordinátor:**
  + Schvalování firem a jejich nabídek
  + Sledování průběhu praxí a přihlášek
  + Export přehledů, archivace a vedení evidence

### 2.3 Omezení produktu

* **Integrace:** Funkčnost systému závisí na dostupnosti a stabilitě API systémů STAG a ARES
* **Časová omezení:** Vývoj v rámci termínů závěrečné práce
* **Bezpečnost:** Nutnost autorizace uživatelů, zabezpečené přenosy dat (HTTPS, šifrování)

### 2.4 Charakteristiky uživatelů

* **Pracovníci katedry:** Pravidelní uživatelé systému, technicky zdatní, spravují nabídky a přihlášky
* **Studenti:** Nejpočetnější skupina, využívají systém zejména v období praxí, potřeba intuitivního UI
* **Organizace:** Externí uživatelé, méně častý přístup, důraz na jednoduchost při správě nabídek

### 2.5 Předpoklady a závislosti

* **ARES API:** Zajišťuje ověření IČO a automatické doplnění údajů o firmě
* **STAG API:** Slouží k autentizaci uživatelů (studentů a zaměstnanců) a získávání jejich údajů
* **Docker:** Systém je kontejnerizován, nasazení vyžaduje prostředí s podporou Docker

### 2.6 Rozdělení požadavků

Požadavky jsou rozděleny podle jednotlivých uživatelských rolí a systémových komponent, což zajišťuje jejich přehlednou evidenci i implementaci. Křížová matice funkcí a komponent bude uvedena v příloze dokumentu.

## 3. Funkční a nefunkční požadavky

## 3.1 Externí rohraní

#### 3.1.1 Uživatelská rozhraní

Uživatelské rozhraní bude navrženo jako intuitivní webové rozhraní. Design vychází z prototypu vytvořeného ve Figma. Bude dodržena jednotná grafická identita univerzity a standardní UI/UX praktiky.

#### 3.1.2 Hardwarová rozhraní

Aplikace bude provozována na serverech univerzity. Komunikace s hardwarem je omezena pouze na běžné vstupní a výstupní zařízení klienta (PC).

#### 3.1.3 Softwarová rozhraní

InternHub se integruje s následujícími systémy: ARES API (ověření IČO) a STAG API (autentizace, uživatelská data). Komunikace probíhá přes REST API s autentizací pomocí JWT.

### 

### 3.1 Funkční požadavky

Níže jsou uvedeny hlavní funkční požadavky systému InternHub, členěné podle uživatelských rolí. Každý požadavek odpovídá scénářům chování ve formátu BDD, které zajišťují jeho testovatelnost.

#### Student

* Zobrazení seznamu aktuálních nabídek praxí
* Detailní náhled na nabídku praxe
* Odeslání přihlášky na konkrétní nabídku
* Odmítnutí nebo stažení přihlášky
* Stažení dokumentu (např. potvrzení o absolvování)
* Úprava osobního profilu

#### Organizace (firma)

* Registrace firmy pomocí IČO s validací přes ARES
* Vytváření nových nabídek praxí
* Úprava a smazání existujících nabídek
* Sledování a správa přihlášek studentů
* Úprava firemního profilu

#### Administrátor / Katedra

* Správa a schvalování registrací firem
* Evidence a export přihlášek a průběhu praxí
* Správa uživatelských účtů

### 3.2 Nefunkční požadavky

#### 3.2.1 Výkon

* Odezva systému na běžný požadavek nesmí překročit 2 sekundy.

#### 3.2.2 Bezpečnost

* Veškerá komunikace bude probíhat pomocí HTTPS.
* Autentizace bude řešena přes STAG, přihlášení bude zabezpečeno JWT.

#### 3.2.3 Spolehlivost

* Systém bude testován pomocí scénářů BDD a akceptačních testů (UAT).
* Modulární architektura umožní omezení dopadu chyb na zbytek systému.

#### 3.2.4 Dostupnost

* Systém bude dostupný alespoň 99 % času během akademického semestru.

#### 3.2.5 Shoda

* Systém bude v souladu s požadavky GDPR.
* Uchovávání dat bude v souladu s předpisy UJEP.

#### 3.2.6 Náklady a termín

* Vývoj systému probíhá v rámci bakalářské práce.
* Nasazení bude realizováno na stávající univerzitní infrastruktuře.
* Termín dokončení: červen 2025.

# 4. Ověření

Každý funkční požadavek bude ověřen prostřednictvím UAT testovacích scénářů. Zkoušky proběhnou na základě uživatelských rolí.

# 5. Přílohy

Příloha A: API reference (STAG, ARES)  
Příloha B: Matice sledovatelnosti