

**I. IDENTIFIKAČNÍ ÚDAJE**

<b>Název práce:</b>	<b>Webová aplikace pro vizualizaci a analýzu meteorologických dat s vyhodnocením rizika koroze</b>
<b>Jméno autora:</b>	<b>Bc. Pavel Šorf</b>
<b>Typ práce:</b>	diplomová
<b>Fakulta/ústav:</b>	Fakulta strojní (FS)
<b>Katedra/ústav:</b>	Ústav přístrojové a řídicí techniky
<b>Vedoucí práce:</b>	Ing. Michal Kuchař
<b>Pracoviště vedoucího práce:</b>	Ústav přístrojové a řídicí techniky

**II. HODNOCENÍ JEDNOTLIVÝCH KRITÉRIÍ**

<b>Zadání</b>	<b>průměrně náročné</b>
<i>Hodnocení náročnosti zadání závěrečné práce.</i>	

<b>Splnění zadání</b>	<b>splněno</b>
Zadání práce je splněno v plném rozsahu.	

<b>Aktivita a samostatnost při zpracování práce</b>	<b>A - výborně</b>
---	--------------------

<b>Odborná úroveň</b>	<b>B - velmi dobře</b>
<p>V některých případech se student dopouští nepřesných, či zavádějících tvrzení, která tak zhoršují dojem z celkové úrovně práce. Zde jsou uvedeny některé příklady.</p> <p>Strany 21 a 28: zbytečné vkládat logo FastApi a React resp.</p> <p>Nepříliš srozumitelný popis databází, strany 31-33.</p> <p>Strana 47: „Pro efektivní komunikaci s databází je nutné vytvořit schéma každé tabulky v databázi pomocí některé z ORM knihoven.“ -&gt; Toto tvrzení je chybné. Vytvoření tabulek a efektivní komunikace jsou dvě různé věci. Tabulky můžete vytvořit i přímým použitím SQL a ne jen pomocí ORM knihoven. ORM knihovny navíc spíše zhoršují efektivitu komunikace s databází, neboť přidávají další vrstvu abstrakce. Z pohledu programátora je to sice jednodušší, ale efektivnější by bylo použít opět SQL.</p> <p>Strana 47 a kód 10: „Ta se definuje v obálce @get.app.“ Použití českého výrazu obálka (bez dalšího vysvětlení) je v tomto kontextu spíše nevhodné. Navíc v Pythonu se tento prvek nazývá decorator (decorator)</p>	

<b>Formální a jazyková úroveň, rozsah práce</b>	<b>A - výborně</b>
<i>Posuďte správnost používání formálních zápisů obsažených v práci. Posuďte typografickou a jazykovou stránku.</i>	

Výběr zdrojů, korektnost citací	A - výborně
---------------------------------	-------------

<b>Další komentáře a hodnocení</b>
V souboru README.md chybí postup pro spuštění celkové aplikace. Dále je škoda, že aplikace není kontejnerizovaná, což by usnadnilo její nasazení.

### III. CELKOVÉ HODNOCENÍ A NÁVRH KLASIFIKACE

Student provedl rešerši nástrojů pro vývoj webových aplikací a to jak pro *backend* i *frontend*. Student dále provedl rešerši pro vyhodnocení rizika koroze v budovách. Na základě této rešerše vyvinul webovou aplikaci pro monitorování meteorologických dat pro vybrané místo pomocí polohy GPS. Na základě meteorologických dat upozorňuje na riziko koroze pomocí jednoduchých pravidel (obvykle překročení kritické hodnoty vybraných veličin). Student vyvinul jak serverovou část, tak i grafické rozhraní pro uživatele. Student pracoval samostatně a prokázal schopnost nastudovat komplexní problém a implementovat řešení.

Předloženou závěrečnou práci hodnotím klasifikačním stupněm **A - výborně**.

Datum: 13.6.2024

Podpis: