

Společná část popisu:

1. Výběr zadání

Vytvořte tříčlenné (preferované), případně dvoučlenné týmy. Vedoucí týmu se přihlásí na jednu z šesti variant. Nejpozději do **1. 11. 2019** se členové týmu ujistí, že spolupráce funguje, nejlépe tím, že začnou na projektu pracovat. Do tohoto termínu lze měnit týmy a zadání. Po tomto datu již změny nejsou možné - projekt by se již pravděpodobně nestihl dokončit. Ostatní problémy v týmu budou řešeny individuálně.

2. Cíle projektu

Cílem projektu je navrhnout a implementovat informační systém s webovým rozhraním pro zvolené zadání jedné z variant. Postup řešení by měl být následující:

1. Analýza a návrh informačního systému (analýza požadavků, tvorba diagramu případů užití, modelu relační databáze)
2. Volba implementačního prostředí - databázového serveru a aplikační platformy
3. Implementace navrženého databázového schématu ve zvoleném DB systému
4. Návrh webového uživatelského rozhraní aplikace
5. Implementace vlastní aplikace

3. Rozsah implementace

Implementovaný systém by měl být prakticky použitelný pro účel daný zadáním. Mimo jiné to znamená:

- Musí umožňovat vložení odpovídajících vstupů.
- Musí poskytovat výstupy ve formě, která je v dané oblasti využitelná. Tedy nezobrazovat obsah tabulek databáze, ale prezentovat uložená data tak, aby byla pro danou roli uživatele a danou činnost užitečná (např. spojit data z více tabulek, je-li to vhodné, poskytnout odkazy na související data, apod).
- Uživatelské rozhraní musí umožňovat snadno realizovat operace pro každou roli vyplývající z diagramu případů použití (use-case). Je-li cílem např. prodej zboží, musí systém implementovat odpovídající operaci, aby uživatel nemusel při každém prodeji ručně upravovat počty zboží na skladě, pamatovat si identifikátory položek a přepisovat je do objednávky a podobně.

Kromě vlastní funkcionality musí být implementovány následující funkce:

- Správa uživatelů a jejich rolí (podle povahy aplikace, např. obchodník, zákazník, administrátor). Tím se rozumí přidávání nových uživatelů u jednotlivých rolí, stejně tak možnost editace a mazání nebo deaktivace účtů. Musí být k dispozici alespoň dvě různé role uživatelů.
- Ošetření všech uživatelských vstupů tak, aby nebylo možno zadat nesmyslná nebo nekonzistentní data.
 - Povinná pole formulářů musí být odlišena od nepovinných.
 - Hodnoty ve formulářích, které nejsou pro fungování aplikace nezbytné, neoznačujte jako povinné (např. adresy, telefonní čísla apod.) Nenuťte uživatele (opravujícího) vyplňovat desítky zbytečných řádků.
 - Při odeslání formuláře s chybou by správně vyplněná pole měla zůstat zachována (uživatel by neměl být nucen vyplňovat vše znovu).
 - Pokud je vyžadován konkrétní formát vstupu (např. datum), měl by být u daného pole naznačen.
 - Pokud to v daném případě dává smysl, pole obsahující datum by měla být předvyplněna aktuálním datem.
 - Nemělo by být vyžadováno zapamatování a zadávání generovaných identifikátorů (cizích klíčů), jako např. ID položky na skladě. To je lépe nahradit výběrem ze seznamu. Výjimku tvoří případy, kdy se zadáním ID simuluje např. čtečka čipových karet v

knihovně. V takovém případě prosím ušetřete opravujícímu práci náповědou několika ID, která lze použít pro testování.

- Žádné zadání nesmí způsobit nekonzistentní stav databáze (např. přiřazení objednávky neexistujícímu uživateli).
- Přihlašování a odhlašování uživatelů přes uživatelské jméno (případně e-mail) a heslo. Automatické odhlášení po určité době nečinnosti.

4. Implementační prostředky

3.1 Uživatelské rozhraní

- HTML5 + CSS, s využitím JavaScriptu, pokud je to vhodné. Je povoleno využití libovolných volně šířených JavaScriptových a CSS frameworků (jQuery, Bootstrap, atd.)
- Případně lze využít i AJAX či pokročilejší klientské frameworky (Angular, React, apod.), není to ale vyžadováno.

Rozhraní musí být funkční přinejmenším v prohlížečích Chrome, Firefox a Internet Explorer.

3.2 Implementační prostředí

- PHP + MySQL (server eva nebo jiný dostupný), případně libovolný open source PHP framework.
- Alternativně jiná serverová technologie (např. Java, .NET, Python, Ruby, apod.) - domluvte se na detailech (viz kontakt níže).

Při použití relační databáze **specifikujte integritní omezení** (např. unikátní hodnoty, cizí klíče, apod.) při vytváření databáze. Neponechávejte zajištění konzistence dat pouze na aplikaci. V MySQL je k tomu třeba použít tabulky typu InnoDB (typ je možno zvolit při vytváření tabulky nebo změnit dodatečně).

Použití jiné platformy je možné, ale je nutné se předem domluvit se cvičícím a samostatně si zajistit umístění vytvořené aplikace, aby bylo možno ji předvést.

5. Dokumentace

Součástí projektu je stručná dokumentace k implementaci, která popisuje, které PHP skripty (případně kontrolery, presentery apod. podle zvoleného frameworku) implementují jednotlivé případy použití. Tato dokumentace je součástí dokumentu [doc.html](#), viz níže.

6. Odevzdání

Odevzdání probíhá přes IS FIT. Od okamžiku odevzdání nejméně do **31. 1. 2020** musí být dále funkční aplikace přístupná přes síť Internet na některém fakultním nebo jiném serveru. Tato aplikace musí umožňovat přihlášení pod všemi rolemi uživatelů a musí být naplněna ukázkovými daty, na kterých lze vyzkoušet všechny funkce. Pokud je to technicky možné, použijte umístění <http://www.stud.fit.vutbr.cz/~xlogin00/IIS>. Vyhnete se případným problémům s funkčností hostingu. Výjimku tvoří předem domluvená a individuálně předvedená řešení na exotických platformách.

Přes IS se odevzdává jeden archiv pojmenovaný vas_login.zip obsahující:

- Všechny zdrojové kódy a datové soubory aplikace. Vzhledem k limitu velikosti odevzdaného souboru ve WISu (2 MB) odevzdávejte pouze vlastní kód a data. Neodevzdávejte prosím kódy použitých knihoven a frameworků třetích stran. Místo toho uveďte pouze jejich verze v dokumentaci.
- SQL skript nebo jiný prostředek pro vytvoření a inicializaci schématu databáze.
- Soubor [doc.html](#) obsahující dokumentaci. Stáhněte si [šablonu dokumentace](#), editujte a přiložte k odevzdanému projektu. Respektujte prosím pokyny obsažené v tomto souboru. Některé pokyny mají formu komentářů v HTML kódu šablony.

Za každý tým odevzdává pouze **vedoucí týmu**.

Termín pro odevzdání do IS FIT je **2. 12. 2019**. Po tomto termínu již nelze projekt akceptovat.

7. Body

Za projekt je možno získat až 30 bodů.

8. Kontakt

Jiří Hynek (ihynek@fit.vut.cz)

Popis varianty:

Úkolem zadání je vytvořit informační systém pro správu a registraci výukových kurzů. Každý výukový kurz má nějaké označení, pomocí kterého ho uživatelé systému budou moci vhodně odlišit a další atributy (popis, typ, tagy, cena, apod.). Výukový kurz je dále reprezentován sadou termínů, lektory, kteří školí studenty a garantem kurzu, který kurz spravuje a potvrzuje registrace studentů. Termín má název, typ (přednáška, cvičení, zkouška, domácí úkol, apod.) popis, možné hodnocení (pro jednoduchost uvažujte celkové hodnocení kurzu 0-100 bodů), datum a volitelně také místnost, kde se konají. Místnosti jsou spravovány vedoucím a mají definovanou polohu (adresa, číslo dveří, apod.), typ (sál, laboratoř), kapacitu, vybavení, apod. V daný časový interval se v místnosti může konat pouze jeden termín. Uživatelé budou moci dále informační systém používat následujícím způsobem:

- **administrátor:**
 - spravuje uživatele
 - má práva všech následujících rolí
- **vedoucí:**
 - schvaluje kurz
 - spravuje místnosti, které jsou přístupné pro výuku kurzu
 - má rovněž práva garanta (může si vytvářet kurzy a schvalovat si je)
- **garant:**
 - zakládá a spravuje kurz (vytváří popis a termíny kurzu)
 - zadává lektory, schvaluje registrace studentů
 - má rovněž práva lektora (může být lektorem svého nebo jiného kurzu)
- **lektor:**
 - modifikuje termíny kurzu, zadává hodnocení studentů
 - volitelně přidává soubory k termínům kurzů (dokumenty, prezentace, obrázky, apod.)
 - má rovněž práva studenta (může si zapsat kurz, který neučí)

- **student:**
 - registruje se na kurzy (schvaluje garant)
 - vidí své hodnocení, může stahovat soubory kurzu, na který je zapsaný
 - vidí svůj rozvrh
 - má rovněž práva neregistrovaného návštěvníka
- **neregistrovaný návštěvník:**
 - vyhledává kurzy, prochází informace o kurzu

Každý registrovaný uživatel má možnost editovat svůj profil.

Typy na možná rozšíření:

- lektor může nastavit práva sdílení souborů (veřejný, pro registrované, pro zapsané)
- lektor může nastavit v termínech odevzdávání studentských souborů (studentských projektů) - typ souboru, regulární výraz jména, maximální velikost apod.