

Inštrukcie, ako by malo vyzerat' správne riešenie diplomovej alebo bakalárskej práce v študijnom programe Hospodárska informatika

Tieto inštrukcie slúžia ako všeobecný návod, ktorý môže byť prispôsobený požiadavkám konkrétnej témy alebo príslušného vedúceho témy. Všetky inštrukcie boli so študentmi prebraté v rámci výučby príslušných predmetov:

- Bakalárske práce: Úvod do hospodárskej informatiky, Analýza a návrh IS, Manažment projektov, Podniková analytika.
- Diplomové práce: Úvod do hospodárskej informatiky, Analýza a návrh IS, Manažment projektov, Podniková analytika, Objavovanie znalostí, Technológie spracovanie veľkých dát, Manažment znalostí, Vývoj a testovanie softvérových systémov.

Inštrukcie sú rozdelené do 3 kategórií podľa najčastejšie sa objavujúcich tém prác. Vo všetkých prípadoch je očakávaným výsledkom práce softvérový výstup:

- Zavedenie nového softvérového riešenia ako odpoveď na identifikovaný problém alebo príležitosť.
- Analýza dátovej vzorky s cieľom riešiť identifikovaný problém, alebo príležitosť.
- Implementácia Big-data technológií.

Zavedenie nového softvérového riešenia ako odpoveď na identifikovaný problém alebo príležitosť

1.fáza – analýza

- Identifikovať problém, hrozbu alebo príležitosť, na ktorý/ú je možné reagovať pomocou vhodného softvérového riešenia.
- Identifikovať cieľovú skupinu, ktorá bude reprezentovať koncového používateľa nového riešenia.
- Ak ide o problém alebo príležitosť na úrovni fungovania organizácie (primerane aj v ostatných prípadoch), je vhodné vykonať procesnú analýzu, ktorej výsledky budú následne použité pri tvorbe základnej predstavy vlastného riešenia. Výsledkom je množina BPMN diagramov, ktoré zrozumiteľne reprezentujú fungovanie organizácie a jasne ukazujú kritické miesta ako potenciál pre nasadenie nového softvérového riešenia za účelom pozitívnej zmeny.
- Vytvoriť základnú predstavu (popis na manažérskej úrovni), ako by mohlo fungovať novo navrhované riešenie. Táto predstava najčastejšie vzniká v intenzívnej komunikácii s cieľovou skupinou, ale nejde do hĺbky na jednotlivé funkcie a spôsob ich fungovania.
- **Jasne definovať merateľné kritéria úspešnosti alebo kľúčové ukazovatele výkonnosti**, ktoré budú v závere použité na vyhodnotenie, či navrhnuté vlastné riešenie splnilo očakávania. Každéj metrike priradiť spôsob jej vyhodnotenia a cieľovú (očakávanú) hodnotu (ak je možné odmerať ju pred zavedením softvérového riešenia, tak je potrebné tak urobiť tiež a získať tak jej referenčnú hodnotu).

- Na základe tejto predstavy realizovať analýzu súčasného stavu v 2 (nájsť a preštudovať vhodné zdroje, kvalitné časopisecké a konferenčné príspevky v práci primeranom počte a štruktúre sú podmienkou) rovinách:
 - Identifikácia a porovnanie konkurenčných riešení s cieľom potvrdiť alebo zamietnuť potenciál pre novú alternatívu (gap). Na ktoré aspekty riešenia je potrebné sa zamerať, aby očakávaný prínos bol čo najvyšší v porovnaní so súčasným stavom.
 - Identifikácia, porovnanie a výber vhodných technológií alebo metód pre tvorbu vlastného riešenia.

2.fáza – návrh

- Vybrať vhodný model životného cyklu, ktorý určí spôsob realizácie celého procesu softvérového vývoja, napr. vodopád, evolučný, prototypový alebo agilný. Tento výber musí byť jasne zdôvodnený a všetky ďalšie kroky musia byť v súlade s vybraným modelom.
- Identifikovať používateľské požiadavky, ktoré vychádzajú zo základnej predstavy a sú výsledkom komunikácie s cieľovou skupinou. Každá požiadavka musí mať jasne pridelenú prioritu, napr. pomocou MoSCoW metódy.
- Identifikovať základných aktérov, ktorí budú navrhované softvérové riešenie používať. Výsledkom je UML diagram prípadov použitia, ktorý jasne a zrozumiteľne priraduje jednotlivým aktérom príslušné funkcie.
- Vytvoriť používateľské scenáre, ktoré zrozumiteľne definujú nielen normálne (typické) spôsoby použitia jednotlivých funkcií, ale aj alternatívne toky a výnimky.
- V prípade použitia objektovo-orientovanej paradigmy vytvoriť UML diagram tried a UML sekvenčné diagramy na základe používateľských scenárov. Všetky vytvorené UML diagramy a používateľské scenáre musia byť navzájom konzistentné.
- Vytvoriť architektúru vlastného riešenia, napr. MVC charakteru, ktorá jasne definuje, z akých úrovní sa bude dané riešenie skladať, aké technológie a metódy budú použité na jednotlivých úrovniach a najmä, ako budú jednotlivé úrovne navzájom interagovať.
- Vytvoriť vizuálne prototypy na rôznych úrovniach (low-fidelity, high-fidelity alebo mockup) pomocou vhodného nástroja ako prvú ukážku vzhľadu používateľského rozhrania a spôsobu jeho fungovania (User Experience). Tento návrh následne konzultovať s cieľovou skupinou (aj opakovane) s cieľom vyhnúť sa neustálemu prerábaniu vo fáze implementácie alebo nespokojnosti vo fáze testovania.

3.fáza – implementácia

- Organizovať fázu implementácie riešenia v súlade s vhodným o súvisí s výberom modelu životného cyklu. Vo všeobecnosti je výsledkom viacerých funkčných prototypov, ktoré sú podrobené príslušnej forme testovania, napr. unit, integračné, systémové alebo akceptačné.

4.fáza – testovanie

- Odporúčame riadiť sa V-modelom, ktorý definuje typy testovania pre všetky úrovne procesu softvérového vývoja.

- Vybrať vhodný typ testovania v závislosti od voľby modelu životného cyklu, vo fáze návrhu (napr. vodopád, prototypový, evolučný alebo agilný).
 - Otestovať správne fungovanie softvérového sa výstupu z pohľadu jeho vnútornej štruktúry a obsahu tradične realizuje počas fázy implementácie.).
 - V prípade modelu vodopád je správnosť fungovania z pohľadu stanovených požiadaviek predmetom testovania po skončení implementácie.
 - V prípade agilného modelu je zákazník priamo zainteresovaný do fázy implementácie, t.j. aj akceptačné testovanie sa realizuje v tejto fáze.
- Na začiatku testovania zadať jeho ciele, scenáre a spôsob vyhodnotenia. Okrem testovania správneho fungovania je potrebné zvážiť aj testy zamerané na výkonnosť, škálovateľnosť, bezpečnosť alebo UX (User Experience).
- V prípade manuálneho testovania prostredníctvom dotazníka odporúčame pri jeho tvorbe použiť Likertovu škálu. Na začiatku študent definuje cieľovú skupinu testovania, dôvody prečo práve takáto cieľová skupina a podmienky testovania. Výsledky dotazníka musia byť jasne, zrozumiteľne a graficky vyhodnotené
- Automatické testovanie je možné realizovať pomocou vhodných nástrojov ako napr. Selenium alebo Robot framework pre webové aplikácie a hybridné mobilné aplikácie. Pre automatizované testovanie Android aplikácií je možné použiť inštrumentačné testy v Android Studiu.

5. fáza – overenie

- V ideálnom prípade nasadiť výsledné riešenie do praxe. Ak to nie je možné, tak aspoň vhodná forma overenia v prevádzke viacerými používateľmi.
- Objektívne odmerať merateľné kritériá úspešnosti stanovených v 1. fáze na porovnanie dosiahnutých výsledkov s kľúčovými ukazovateľmi výkonnosti. Na základe toho vyhodnotiť, či navrhnuté vlastné riešenie splnilo očakávania a odmerať jeho potenciálne prínosy.
- Pre študentov HI je veľmi dôležité, aby dokázali jasne identifikovať a odmerať prínosy vytvoreného softvérového riešenia (informačného systému). Napr. na úrovni ekonomických ukazovateľov, zefektívnenia podnikových procesov, vytvorenia inovatívnych služieb, zlepšenie používateľského zážitku a pod.

Analýza dátovej vzorky s cieľom objaviť nové, potenciálne užitočné znalosti.

1.fáza – Pochopenie cieľa

- Na základe charakteru vybraných dát realizovať analýzu súčasného stavu (nájsť a preštudovať vhodné zdroje, kvalitné časopisecké a konferenčné príspevky v primeranom počte a štruktúre sú podmienkou):
 - Identifikovať a porovnať relevantné alternatívy riešenia s cieľom potvrdiť alebo zamietnuť potenciál pre novú alternatívu (gap). Identifikovať, na ktoré aspekty riešenia je potrebné sa zamerať, aby očakávaný prínos bol čo najvyšší v porovnaní so súčasným stavom, napr. také formulovanie úlohy ktoré ešte nebolo riešené, resp. bolo riešené ale inými metódami a s neuspokojivou kvalitou výsledku a pod.

- Identifikovať, porovnať a vybrať vhodné technológie a metódy pre tvorbu vlastného riešenia. Tu je nevyhnutné nájsť väčší počet vedeckých článkov, ktoré sa čo najbližšie zameriavajú na identifikovanú úlohu, resp. riešia podobnú úlohu v inom kontexte a pod.
- Jasne a správne stanoviť viacero úloh DM, ktoré je potrebné preskúmať a realizovať (súčasťou je napr. zvoliť napr. cieľový atribút pre úlohy klasifikačného typu) - stanoviť hypotézu (hypotézy).
- **Jasne definovať merateľné kritéria úspešnosti alebo kľúčové ukazovatele výkonnosti**, ktoré budú v závere použité na vyhodnotenie, či navrhnuté vlastné riešenie splnilo očakávania. Každéj metrike priradiť spôsob vyhodnotenia – stanoviť tézy

2. fáza – Pochopenie dát

- Realizovať exploratívnu alebo štatistickú analýzu dát s cieľom pochopiť jednotlivé atribúty a potenciálne vzťahy medzi nimi.
 - V prípade, že rozdelenie hodnôt daného atribútu nie je totožné s normálnym, tak sa zamerať na neparametrické štatistické metódy.
 - Analyzovať nielen vzťahy medzi vstupnými atribútmi navzájom, ale aj medzi nimi a cieľovým atribútom.
 - Analyzovať a identifikovať prípadnú multikolinearitu, na základe ktorej následne upraviť množinu atribútov pre fázu modelovania.

3.fáza – Predspracovanie dát

- Aplikovať výsledky predchádzajúcej fázy s cieľom pripraviť vzorku dát na aplikáciu vybraných metód strojového učenia vo fáze modelovania. Zdokumentovať všetky uskutočnené operácie, aby bolo možné spätne vyhodnotiť, aký mali vplyv na dosiahnutý výsledok.
- Dôraz je potrebné klásť aj na vhodný a odôvodnený výber atribútov (feature selection), aj s prihliadnutím na cieľovú úlohu (pozri napr. prednáška o zhlukovaní resp. klasifikácii v predmete OZ – filtračné, obáľkové, príp. vložené modely pre výber atribútov)
- V prípade nerovnomernej distribúcie hodnôt cieľového atribútu zvážiť vhodné metódy na jej odstránenie vzhľadom na dosiahnuté výsledky vo fáze modelovania (napr. podvzorkovanie, nadvzorkovanie, nákladová matica, pozri aj riešenie náročnejších klasifikačných úloh v druhej prednáške venovanej klasifikácii v predmete OZ).

4. fáza – Modelovanie

- Vybrať vhodné metódy a svoj výber podrobne zdôvodniť:
 - napr. v prípade klasifikácie lineárne, resp. nelineárne klasifikátory, pravdepodobnostné alebo ďalšie typy podľa charakteru riešenej úlohy a požadovaného výstupu.
 - v prípade zhlukovania zdôvodniť výber metriky podobnosti, resp. vzdialenosti a typ zhlukovacej metódy, podobne aj v prípade úlohy detekcie anomálií.
 - zvážiť použitie rozšírených klasifikačných metód ako sú semi-kontrolované učenie, aktívne učenie, resp. zloženú klasifikáciu (učenie súborom metód).

- zamyslieť sa nad rozsahom skreslenia (bias) a rozptylu (variance) pri vybraných modeloch a optimalizovať ich nastavenia v zmysle najvhodnejšieho kompromisu medzi skreslením a rozptylom s prihliadnutím na cieľovú aplikáciu.
- **Vybrať vhodný spôsob vyhodnotenia:**
 - napr. holdout, krížová validácia alebo bootstrap v prípade klasifikačnej úlohy.
 - krivka presnosť-návratnosť, resp. ROC krivka v prípade výsledku klasifikácie vo forme numerického skóre.
 - vhodné interné a/alebo externé kritériá pre validáciu zhukov v prípade zhukovania.

5.fáza – Vyhodnotenie

- Objektívne odmerať **merateľné kritéria úspešnosti stanovené v 1. fáze na porovnanie dosiahnutých výsledkov s kľúčovými ukazovateľmi výkonnosti**. Na základe toho vyhodnotiť, či navrhnuté vlastné riešenie splnilo očakávania a odmerať jeho potenciálne prínosy (napr. aj voči referenčným prácam). Na záver prijať alebo zamietnuť stanovené hypotézy.

6.fáza – Nasadenie

- V ideálnom prípade nasadiť vytvorenú aplikáciu, resp. model. Ak nie je možné, aspoň vhodná forma overenia v prevádzke viacerými používateľmi. Ak to nie je možné, tak aspoň detailná predstava o spôsobe a forme nasadenia modelu v reálnej praxi a presných spôsoboch hodnotenia jeho kvality a úspešnosti.
- Pre študentov HI je veľmi dôležité, aby dokázali jasne identifikovať a odmerať prínosy vytvoreného modelu (modelov). Napr. na úrovni ekonomických ukazovateľov, zefektívnenia podnikových procesov, zobjektívnenia rozhodovacieho procesu a pod.

Implementácia Big-data technológií

1.fáza – Analýza

- Identifikovať a popísať problém, ktorý bude riešený v rámci záverečnej práce (napr. problém dátovej integrácie, spracovania prúdových dát, nasadenia prediktívnych modelov a pod.). Uviesť príklady oblastí využitia, v ktorých sa daný problém najčastejšie vyskytuje.
- Pre zvolenú doménu a dáta navrhnuť a popísať ukázkové používateľské scenáre, na ktorých má byť riešenie daného problému testované/demonštrované. Scenáre by mali popisovať typ a charakteristiku vstupných dát na požadovaný výstup spracovania, resp. požadovanú funkčnosť potrebnú pre riešenie daného problému.
- Pre každý testovací scenár je potrebné definovať sledované kritériá kvality v podobe merateľných metrik, na základe ktorých sa bude dané riešenie vyhodnocovať (napr. pamäťová alebo časová náročnosť výpočtu, latencia spracovania, presnosť výstupov a pod.)

2.fáza – Návrh riešenia

- Na základe definovaných používateľských scenárov navrhnúť architektúru riešenia daného problému. Vhodnou metódou je napr. dekompozícia systému na komponenty s presne definovanou funkčnosťou, ktoré je možné implementovať existujúcimi technológiami na spracovanie veľkých dát (napr. fronty správ, distribuované databázy, rámce pre paralelné a distribuované výpočty a pod.). Navrhnutú architektúru popísať vhodným spôsobom, napr. pomocou blokových schém alebo UML diagramov.
- Na základe navrhnutej architektúry analyzovať existujúce technológie vhodné na jej implementáciu a vypracovať ich prehľad. V rámci prehľadu stanoviť kritériá pre porovnanie jednotlivých technológií ako napr. softvérové a hardvérové požiadavky, podporované operácie nad dátami, štandardy výmeny dát, dopytovacie jazyky, podpora programovacích jazykov a prostredí, požiadavky na škálovateľnosť, konzistentnosť, licenčné podmienky atď.
- Vybrať vhodné technológie pre implementáciu architektúry a zdôvodnenie ich výberu na základe stanovených kritérií. Ak je cieľom práce porovnať rôzne technológie, vybrať jednotlivých kandidátov, ktorí budú testovaní v rámci záverečnej práce.

3.fáza – Implementácia a nasadenie riešenia

- Na vhodnej úrovni detailov popísať implementáciu navrhnutej architektúry pomocou zvolených technológií. Implementácia môže byť rozdelená do viacerých iterácií, napr. podľa testovacích scenárov, resp. podľa testovaných kandidátov pre jednotlivé technológie.
- Do popisu zahrnúť aj implementáciu pomocných nástrojov ako napr. rôznych generátorov dát, klientov pre automatické testovanie, nástrojov pre transformovanie formátu dát, nástrojov pre monitorovanie a spracovanie logov, a pod.
- Popísať testovacie hardvérové a softvérové prostredie, v ktorom bude navrhnutá implementácia testovaná a spôsob nasadenia riešenia do tohto prostredia (napr. pomocou UML diagramov nasadenia).

4.fáza – Testovanie a vyhodnotenie

- Realizovať testovacie scenáre pomocou implementovanej architektúry. V rámci každého scenára je možné definovať parametre a ich rozsah, ktoré sa budú pri testovaní meniť (napr. veľkosť dát, rýchlosť aktualizácií, parametre softvérového a hardvérového prostredia a pod.)
- Pre danú kombináciu parametrov namerať zvolené metriky pre každý testovací scenár.
- Vhodne porovnať namerané výsledky a vyhodnotiť navrhnuté riešenie, resp. porovnávané alternatívy.