

## **Dokumentácia k inžinierskemu dielu**

Tím DUtí(m).

<b>Členovia tímu:</b>	Bc. Boris Slíž Bc. Filip Pavkovček Bc. Jozef Melicherčík Bc. Marek Pacher Bc. Michal Sojka Bc. Tomáš Bende Bc. Veronika Búcsiová Mgr. Veronika Horniczka
<b>Vedúci projektu:</b>	Ing. Ján Lang, PhD.
<b>Tím:</b>	č.18 [FIIT DU]
<b>Akademický rok:</b>	2018/2019
<b>Dátum poslednej úpravy:</b>	23.11.2018

# Obsah

<b>1 Úvod</b>	<b>2</b>
<b>2 Globálne ciele projektu na zimný semester</b>	<b>3</b>
<b>3 Celkový pohľad na systém</b>	<b>4</b>
3.1 Architektúra	4
3.2 Dátový model	6
Tabuľky modelu	6
3.3 Moduly	12
3.3.1 Používateľské roly	12
3.3.2 Informatívne testovanie	13

# 1 Úvod

Svet okolo nás sa neustále vyvíja. Avšak v poslednej dobe si ľudia čoraz viac uvedomujú, že spôsob, akým sa deti a následne aj dospelí vzdelávajú, je zastaralý. Spôsob dnešnej výučby sa vyvinul ešte počas priemyselnej revolúcie, keď boli požiadavky na populáciu iné, ako v dnešnej dobe. U detí sa oceňovali iné vlastnosti a učili ich iným schopnostiam a návykom, ktoré im v minulosti pomohli zaradiť sa do pracovného života v továrňach. Avšak v dnešnej dobe sa čoraz viac kladie dôraz na individualitu. To však učebné osnovy nedovoľujú.

Základom vzdelávania je komunikácia a najpoužívanejším prostriedkom na komunikáciu v modernej spoločnosti je web. Webové technológie ponúkajú dostatočný priestor na vytvorenie nástroja, ktorý by podporil prepojenie požiadaviek, ktoré majú na žiakov vyučujúci, a požiadaviek, ktoré majú zamestnávateľia na svojich budúcich zamestnancov.

Cieľom nášho projektu je teda navrhnúť a implementovať nástroj, ktorý by prispôbil výučbu moderným požiadavkám, ktoré kladú na svojich budúcich zamestnancov zamestnávateľia. Naším zámerom je pomocou nášho nástroja poskytnúť žiakom potrebné vedomosti v konkrétnych oblastiach. Výučba by sa realizovala pomocou otázok a úloh. Pedagógovia si budú môcť vytvárať vlastné učebné plány, vytvárať hodiny, a úlohy. Bude môcť zdieľať tieto materiály so svojimi žiakmi, ale aj kolegami. A skrz tento nástroj bude vidieť štatistiky o svojich študentoch, poprípade ich slabiny, na ktoré sa bude môcť individuálne zamerať.

Dokument k inžinierskemu dielu obsahuje hlavne technickú stránku projektu. Hlavným cieľom tohto dokumentu je priblížiť čitateľom stav nášho tímového projektu po prvých troch šprintoch. Priblížime ciele, ktoré sme si stanovili, a naznačíme, ako budeme pokračovať v letnom semestri. V tretej kapitole je popísaná architektúra a jednotlivé moduly systému.

## 2 Globálne ciele projektu na zimný semester

S vlastníkom produktu sme si ako hlavné ciele semestra zadefinovali určenie a implementáciu používateľských rolí a možnosť informatívne sa testovať. Taktiež bude potrebné mať možnosť pridať a zobrazit' nápovedy k jednotlivým otázkam v teste. Je veľmi dôležité, aby boli tieto časti finalizované na konci semestra, keďže sa tento systém plánuje použiť na informatívne vzdelávanie študentov FIIT na predmete OOP v ďalšom semestri.

Keďže sa jedná už o dopĺňujúcu funkcionality k existujúcemu projektu, tak ako prvé je nevyhnutné rozšíriť už existujúci dátový model o nové tabuľky a následne tieto tabuľky a ich prepojenia implementovať. Po rozšírení databázy je možné postúpiť na implementáciu nasledovných modulov:

- používateľské roly
  - anonym
  - študent
  - autor
  - učiteľ
  - komisár
  - administrátor
- informatívne testovanie
- vytvorenie, prehliadanie a vymazanie nápovedy

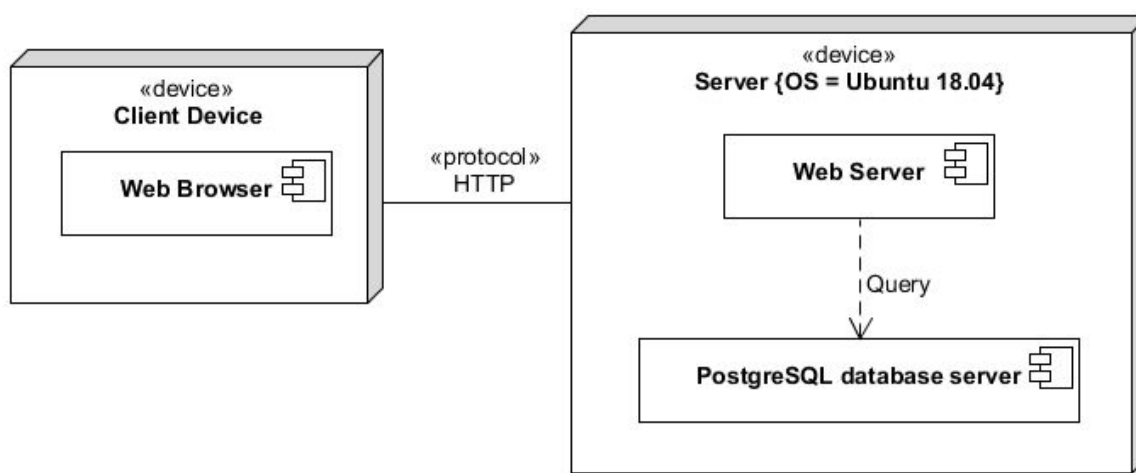
Tieto moduly sú totožné s User stories, ktoré definoval Product Owner. Pre každú rolu je definované aké má právomoci a čo všetko jej nie je umožnené.

Na konci zimného semestra by teda mal systém rozlišovať používateľské roly a ich právomoci. Každý registrovaný používateľ by mal mať možnosť informatívne sa testovať na jednotlivých jednotkách a pridávať a prehliadať nápovedy pri udalostiach.

## 3 Celkový pohľad na systém

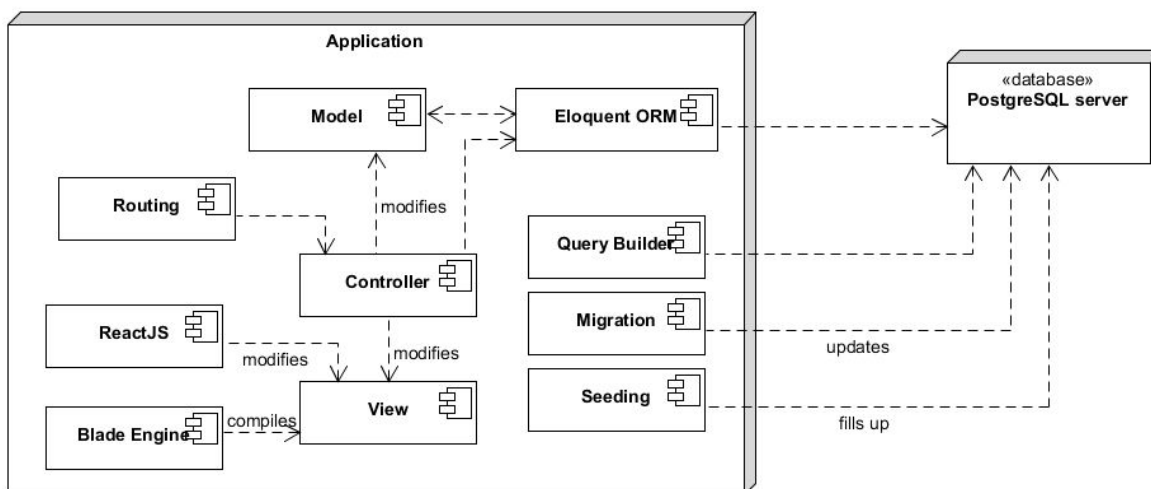
### 3.1 Architektúra

Vyvíjaný systém je webovou aplikáciou architektúry klient-server. Základnými súčasťami sú programovací jazyk PHP verzie 7 a framework Laravel verzie 5, ktorý výrazne obohacuje prácu s jazykom PHP.



Obr. 1: Diagram klient-server architektúry projektu Dilema.

Samotnú architektúru webovej aplikácie môžeme ešte rozdeliť na samostatné komponenty. Ich štruktúra využíva architektonický štýl Model-View-Controller (MVC). Tieto základné MVC komponenty dotvárajú ďalšie pomocné komponenty, ako komponenty na sprostredkovanie dopytov medzi aplikáciou a databázou (Eloquent, Query Builder) a komponenty kompilujúce a modifikujúce používateľské rozhranie (Blade, ReactJS). Dáta sú uskladňované v databáze systému PostgreSQL.



Obr. 2: Diagram komponentov webovej aplikácie projektu Dilema.

Routing spracováva prichádzajúce HTTP požiadavky. Na základe adresy požiadavky vyvolá akciu implementovanú v Controlleri. Taktiež rieši HTTP požiadavky rôznych druhov.

Controller implementuje spracovanie biznis funkcionality aplikácie. Zobrazuje používateľské rozhranie klientovi, autentifikuje a autorizuje používateľa, presmeruje ho a aktualizuje model.

Model reprezentuje dáta v databázovom systéme. Controller môže využívať tieto dáta na aplikovanie aplikačnej logiky.

View reprezentuje frontend časť aplikácie. Oddeľuje prezentačnú logiku od biznis logiky a definuje používateľské rozhranie. Toto používateľské rozhranie je popísané v jazyku PHP, no dotvorené o stavebné štruktúry a špeciálne dopyty pomocou laravelovského enginu Blade, a o dodatočnú JavaScript funkcionality pomocou knižnice ReactJS.

Eloquent ORM slúži na objektovo-relačné mapovanie. Vytvára, aktualizuje a odstraňuje inštancie elementov modelu. Je súčasťou frameworku Laravel.

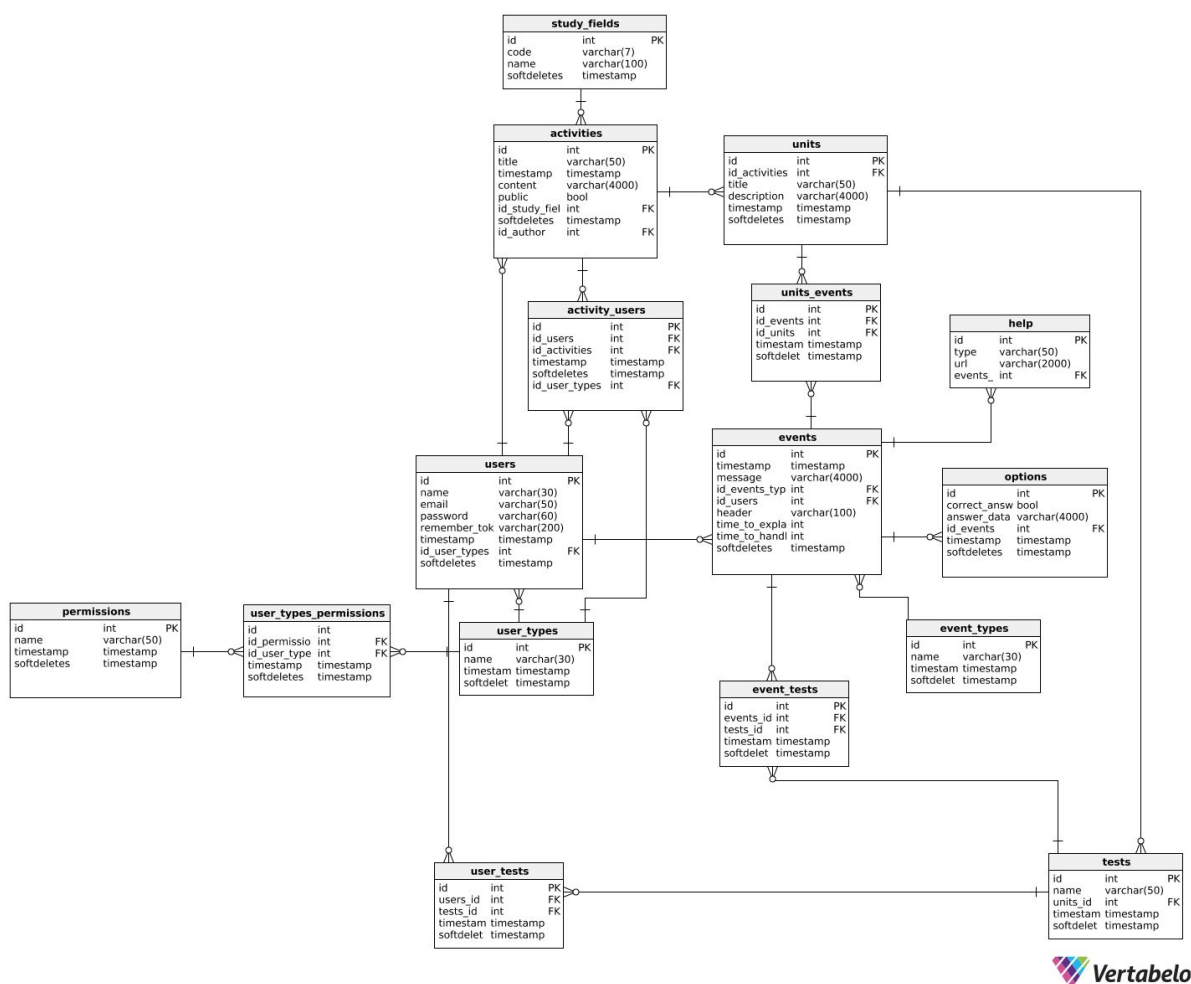
Query Builder výrazne zjednodušuje vytváranie dopytov do databáze. Výsledkom dopytu je dátová štruktúra obsahujúca vyžiadané informácie. Nahrádza dopytovanie sa do databáze štandardnými SQL dopytmi a abstrahuje tak od konkrétneho použitého relačného jazyka.

Migration umožňuje definovať štruktúru databázy v jazyku PHP. Výrazne tak zjednodušuje zásahy do databáze, a tým aj jej verziovanie a prípadné refaktorizačné zásahy, ktoré by inak vyžadovali zložité zmeny štruktúry.

Seeding slúži na naplnenie databáze testovacími, vopred danými dátami. Toto umožňuje jednoduché a efektívne testovanie s použitím týchto dátach.

## 3.2 Dátový model

V tejto časti sa nachádza dátový model a jeho podrobnejší opis. Keďže je náš projekt rozšírením existujúceho projektu, z toho dôvodu už niektoré časti modelu existovali a niektoré sme rozšírili.



Obr. 3: Dátový model projektu Dilema.

## Tabuľky modelu

### Tabuľka users

Táto tabuľka slúži pre ukladanie registrovaných používateľov. Toto im umožní následnú možnosť sa prihlásiť a viesť vlastný účet v systéme s ich údajmi.

### Atribúty:

- *name* [varchar] : meno používateľa
- *email* [varchar] : unikátny reťazec vo forme emailu
- *password* [varchar] : zakryptované heslo
- *remember\_token* [varchar] : informácia o poslednom prihlásení, pre možnosť odhlásenia pri dlhodobej nečinnosti
- *id\_user\_types* [integer] : cudzí kľúč, ktorý sa odkazuje na tabuľku typov používateľa

### Tabuľka *user\_types*

Táto tabuľka definuje typy používateľov, podľa ktorých sa im budú v iných častiach určovať právomoci.

### Atribúty:

- *name* [varchar] : názov používateľského typu

### Tabuľka *events*

V tejto tabuľke sa definujú udalosti k jednotlivým jednotkám. Každá udalosť môže mať viacero odpovedí(možností), kde sa definuje, či je odpoveď, ktorá je priradená ku konkrétnej udalosti správna, alebo nesprávna.

### Atribúty:

- *message* [varchar] : obsah aktivity, ktorý je reprezentovaný textom, tento text krátko popisuje aktivitu
- *header* [varchar] : názov udalosti
- *time\_to\_explain* [integer] : čas, ktorý sa určuje na vysvetlenie udalosti
- *time\_to\_handle* [integer] : čas, ktorý je poskytnutý študentovi na vypracovanie udalosti
- *id\_event\_types* [integer] : cudzí kľúč, ktorý sa odkazuje na tabuľku typov udalostí
- *id\_users* [integer] : cudzí kľúč, ktorý sa odkazuje na tabuľku používateľov

### Tabuľka *event\_types*

Každá udalosť musí mať priradený typ, ktorý je určený v tejto tabuľke. Jeden z možných typov je dichotomická(jedna správna odpoveď) a druhý je polytomická(viacero správnych odpovedí)

### Atribúty:

- *name* [varchar] : názov typu udalosti(dichotomická, polytomická)

### Tabuľka *options*



V tejto tabuľke sa nachádzajú možnosti pre jednotlivé udalosti. Ku každej udalosti je viacero možných odpovedí, ktoré sú označené za správne, alebo nesprávne.

**Atribúty:**

- *answer\_data* [varchar] : možnosť odpovedať textom, ktorý sa napíše ako markdown reťazec
- *correct\_answer* [boolean] : definícia, či je táto konkrétna odpoveď správna, alebo nesprávna
- *id\_events* [integer] : cudzí kľúč, ktorý sa odkazuje na udalosť, ku ktorej odpoveď patrí

**Tabuľka *activities***

V tejto tabuľke sú reprezentované aktivity, ktoré môžu vyjadrovať napríklad predmet. K aktivite sa viažu následne jednotky, ktoré môžu byť považované za vyučovacie hodiny.

**Atribúty:**

- *title* [varchar] : titulok aktivity(jej názov)
- *content* [varchar] : popis predmetu v markdown formáte
- *public* [boolean] : definícia, či je predmet verejný, alebo nie
- *id\_study\_field* [varchar] : definícia oboru
- *id\_author* [integer] : väzba na autora aktivity

**Tabuľka *units***

Do tejto tabuľky sa ukladajú jednotky, ktoré sa považujú za vyučovacie hodiny. Tieto jednotky musia vždy patriť k niektorej aktivite. Pri aktivite je tiež možnosť priamo vytvoriť jednotku.

**Atribúty:**

- *title* [varchar] : titulok jednotky(jej názov)
- *description* [varchar] : krátky a jasný popis jednotky
- *id\_activities* [integer] : cudzí kľúč, ktorý sa odkazuje na aktivitu

**Tabuľka *units\_events***

Toto je väzobná tabuľka medzi udalosťami a jednotkami, kde je vzťah N ku N. To znamená, že udalosť môže patriť k viacerým jednotkám a jednotka môže mať väčší počet udalostí, ktoré sú k nej priradené.

**Atribúty:**

- *id\_units* [integer] : cudzí kľúč, ktorý sa odkazuje na jednotku
- *id\_events* [integer] : cudzí kľúč, ktorý sa odkazuje na udalosť

**Tabuľka *study\_fields***

Táto tabuľka reprezentuje študijný odbor aktivity, ktorý je reprezentovaný kódom.

**Atribúty:**

- *name* [varchar] : názov študijného odboru
- *code* [varchar] : kód študijného odboru

**Tabuľka *activity\_users***

Tabuľka reprezentujúca vzťah N ku N pre aktivity a jej používateľov.

**Atribúty:**

- *id\_users* [integer] : cudzí kľúč, ktorý sa odkazuje na používateľa
- *id\_activities* [integer] : cudzí kľúč, ktorý sa odkazuje na aktivitu
- *id\_user\_types* [integer] : cudzí kľúč, ktorý sa odkazuje na typ používateľa v danej aktivite

**Tabuľka *permissions***

Ukazovateľ právomocí pre jednotlivé typy používateľov.

**Atribúty:**

- *name* [varchar] : názov právomoci

**Tabuľka *user\_types\_permissions***

Väzobná tabuľka umožňujúca multiplikatívny vzťah N:N medzi typmi používateľov a ich právomocami. K jednej právomoci môžu byť viaceré typy používateľov a rovnako k typom môže byť viac právomocí.

**Atribúty:**

- *id\_user\_types* [integer] : cudzí kľúč odkazujúci na ID daného typu používateľa

- *id\_permissions* [integer] : cudzí kľúč odkazujúci na ID danej právomoci

### **Tabuľka *tests***

Táto tabuľka zabezpečuje uloženie testu, ktorý sa bude skladať z viacerých udalostí a bude môcť prislúchať viacerým používateľom, ktorý sa budú môcť otestovať.

#### **Atribúty:**

- *name*[varchar] : názov testu
- *units\_id*[integer] : cudzí kľúč odkazujúci na tabuľku danej jednotky, ktorej prislúcha tento test

### **Tabuľka *event\_tests***

Väzobná tabuľka umožňujúca multiplikatívny vzťah N:N medzi testami a ich udalosťami. K jednému testu môže byť priradených viac úloh a úlohy môžu patriť k rôznym testom.

#### **Atribúty:**

- *events\_id*[integer] : cudzí kľúč odkazujúci na ID danej udalosti
- *tests\_id*[integer] : cudzí kľúč odkazujúci na ID daného testu

### **Tabuľka *user\_tests***

Väzobná tabuľka umožňujúca multiplikatívny vzťah N:N medzi testami a ich používateľmi. K jednému testu môže byť priradených viac používateľov a používatelia môžu patriť k rôznym testom.

#### **Atribúty:**

- *users\_id*[integer] : cudzí kľúč odkazujúci na ID daného používateľa
- *tests\_id*[integer] : cudzí kľúč odkazujúci na ID daného testu

### **Tabuľka *help***

Tabuľka, ktorá definuje typ a odkaz na konkrétnu nápovedu. Každá nápoveda je priamo napojená na udalosť, ku ktorej patrí.

**Atribúty:**

- *type*[varchar] : typ nápovedy
- *url*[varchar] : url adresa na nápovedu k udalosti
- *events\_id*[integer] : cudzí kľúč odkazujúci na ID danej udalosti

## 3.3 Moduly

V tejto časti dokumentácie sú opísané moduly nášho projektu.

### 3.3.1 Používateľské roly

#### **Analýza**

Vyvíjaná webová aplikácia by mala odzrkadlovať rôzne roly, ktoré sa nachádzajú či už v škole, alebo univerzite. Medzi jednoznačne identifikovateľné patrí rola študent a učiteľ, avšak pre bezproblémové fungovanie systému je nutné implementovať aj iné používateľské roly, ktoré pomôžu pri škálovateľnosti v budúcnosti. Tieto roly budú mať rôzne právomoci, ktoré vyplývajú z ich authority.

#### **Návrh**

Navrhnuté roly a ich právomoci:

- Anonym
  - môže prezerať prezentáciu stránky.
  - môže sa registrovať.
- Študent
  - môže byť zahrnutý do predmetu.
  - môže byť testovaný informatívne aj merateľné, ale musí byť zahrnutý v nejakom predmete.
  - môže prehliadať materiály viazané k udalostiam.
  - môže pridávať nápovedy k udalostiam.
  - môže hodnotiť materiály, nápovedy, kurz.
  - môže vytvoriť udalosť, potom sa stáva Autorom.
- Autor
  - môže všetko to čo, Študent.
  - môže vytvárať vzdelávacie aktivity(predmet) a vzdelávacie jednotky(cvičenie, prednáška, seminár, hodina, ...), potom sa stáva Učiteľom.
  - môže upravovať aktivity a jednotky, ktoré vytvoril.
  - môže zmazať aktivity, ktoré vytvoril, kým ich nezvalidoval Komisár.
- Učiteľ
  - môže všetko to, čo Autor.
  - môže prizvať Študentov do svojho predmetu.
  - môže vidieť štatistiku predmetu.
  - môže vidieť štatistiku študentov v jeho predmete.
- Komisár
  - môže všetko to, čo Autor.

- môže validovať aktivitu.
- môže vymazať nápovedu.

## **Implementácia**

Pri súčasnom stave projektu boli implementované nasledovné roly a právomoci:

- Anonym - celá
- Študent
  - môže byť zahrnutý do predmetu.
  - môže byť testovaný informatívne.
  - môže vytvoriť udalosť, potom sa stáva Autorom.
- Autor
  - môže všetko to čo, Študent.
  - môže vytvárať vzdelávacie aktivity(predmet) a vzdelávacie jednotky(cvičenie, prednáška, seminár, hodina, ...), potom sa stáva Učiteľom.
  - môže upravovať aktivity a jednotky, ktoré vytvoril.
  - môže zmazať aktivity, ktoré vytvoril.
- Učiteľ
  - môže všetko to, čo Autor.
  - môže prizvať Študentov do svojho predmetu.
- Komisár
  - môže všetko to, čo Autor.

### **3.3.2 Informatívne testovanie**

#### **Analýza**

Študenti by mali mať možnosť otestovať si svoje schopnosti nie len formou hodnoteného testu, ale aj formou testu, ktorý bude mať len informatívny charakter. Toto testovanie študentom ukáže v akej miere sa oboznámili s učivom danej udalosti a či sú pripravený na hodnotený test. Pri vyberaní odpovede na položenú otázku im aplikácia povie, či bola študentom označená odpoveď správna, alebo nesprávna.

#### **Návrh**

Dôležitým bodom návrhu je grafické rozhranie. Musí byť jednoduché, jednoznačné a intuitívne pre rôznych používateľov. Pod tým sa rozumie, že prostredie nesmie byť odstrašujúce pre začiatočníkov, a má byť aj pre nich intuitívne navigovanie sa v prostredí testovania. Taktiež dôležitým bodom je, že aplikácia musí byť schopná informovať študenta o tom, či jeho odpoveď bola správna alebo nie.

## **Implementácia**

Študentovi sa po prihlásení do webovej aplikácie v hornom paneli zobrazí okrem iného aj možnosť “Testovanie”, kde si zo zoznamu môže vybrať z možností “Informatívne” a “Hodnotený”. Následne je študent presmerovaný na zoznam dostupných informatívnych

testov. Po vybratí testu študent vidí koľko otázok je v teste a vie sa medzi nimi preklikávať. Po označení odpovede sa študent dozvie či je jeho zvolená bola správna (zelená - *Obr.4*) alebo nesprávna (červená - *Obr.5*). K informatívnemu testovaniu sa študent dostane, aj keď klikne na vzdelávaciu aktivitu a zobrazí sa mu zoznam jednotiek. Pri týchto vzdelávacích jednotkách je ikona modrého otáznika, ktorého pomocou sa tiež spustí informatívne testovanie.

Pôvodná trieda sa nazýva ..... a nová trieda je .....

rodič, potomok

zla odpoved 2

zla odpoved 1

Ukončiť test
Predchádzajúca
1
2
3
4
5
Ďalšia

*Obr. 4: Výber správnej odpovede*

Čo môžu dediť objekty?

zla odpoved 2

zla odpoved 1

atribúty a metódy

Ukončiť test
Predchádzajúca
1
2
3
4
5
Ďalšia

*Obr. 5: Výber nesprávnej odpovede*

#### Zoznam jednotiek

Dedenie

?

Polymorfizmus

?

*Obr. 6: Modrá ikona na spustenie informatívneho testovania*