

UNIVERSITETI I PRISHTINËS “HASAN PRISHTINA”
FAKULTETI I SHKENCAVE MATEMATIKO-NATYRORE



Tema: Exam Simulator
Lënda: OOP dhe GUI

Grupi 7:

Andi Alidema

Syard Dauti

Profesor:

Armend Salihu

Përmbajtaj:

1.Definimi i detyrës

2.Interferimi i userit me aplikacionin

3.Arkitektura e aplikacionit

4.Rëndesia e aplikacionit

1. Definimi i detyrës:

Kërkesa e detyrës për këtë projekt është ndërtimi i një aplikacioni i cili i lejon userit futjen e pyetjeve në databazë në kuadër të një shablloni përkates në të cilin perfshihet edhe numri i pikeve, si input i detyrueshëm. Aplikacioni mbanë logjiken e nevojshme ashtu që pytjet të cilat pas një kohe nuk shihen të përshtateshme për përdoruesin ato mund të hiqen nga databaza. Sipas preferencave aplikacioni mundëson shfaqjen e të gjitha pyetjeve të cilat i ka vendosur useri dhe pjesa më rëndësishme e aplikacionit është gjenerimi i testit me pyetje random me kushte që numri i tërësishëm i pikëve për pyetje të jetë total 100.

2. Interferimi i userit me aplikacionin

Me ekzekutimin e ikonës për startimin e aplikacionit, userit u hapet dritarja siq shihet në figurën 2.1.

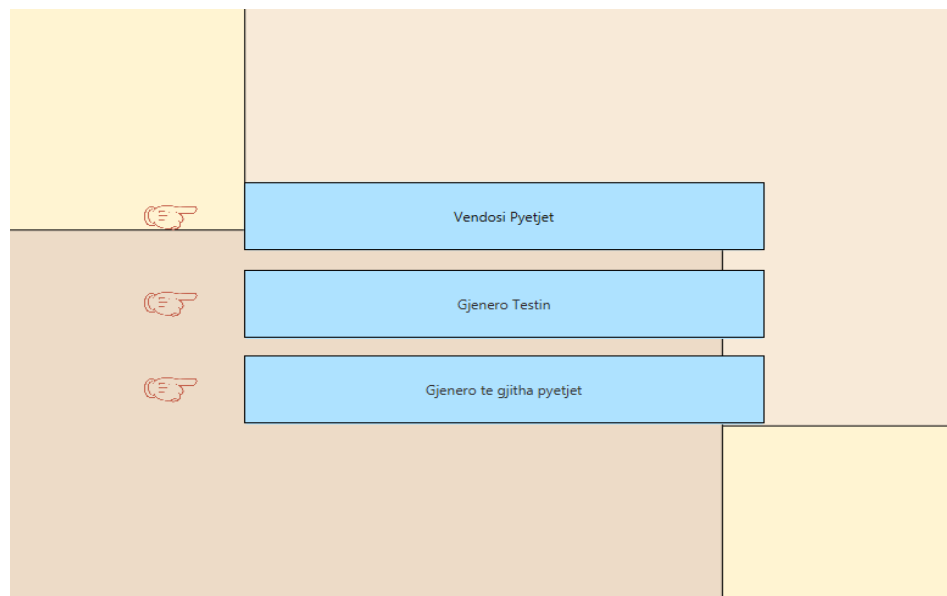
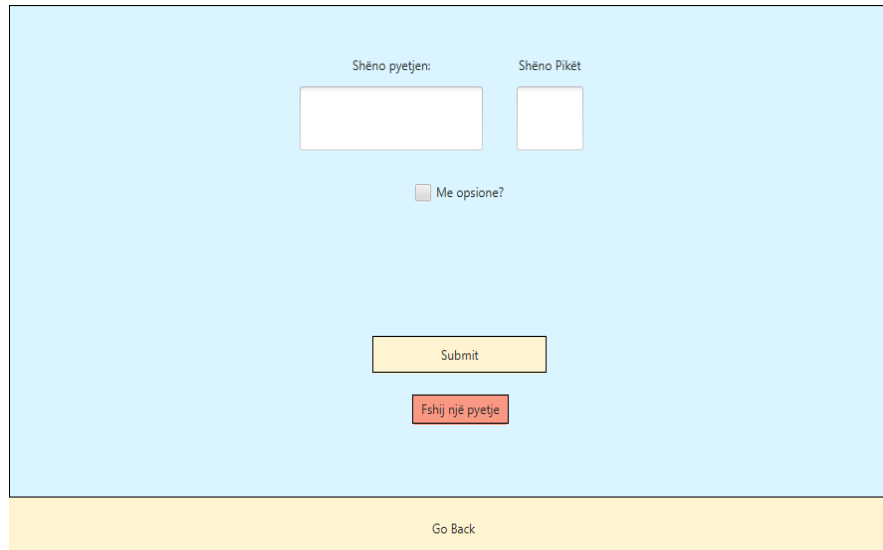


Figura 2.1: Dritarja komanduese

Nga figura shohim se në panel janë të vendosur 3 butona me emërtime përkatëse:

2.1 Butoni “Vendosi pyetjet”

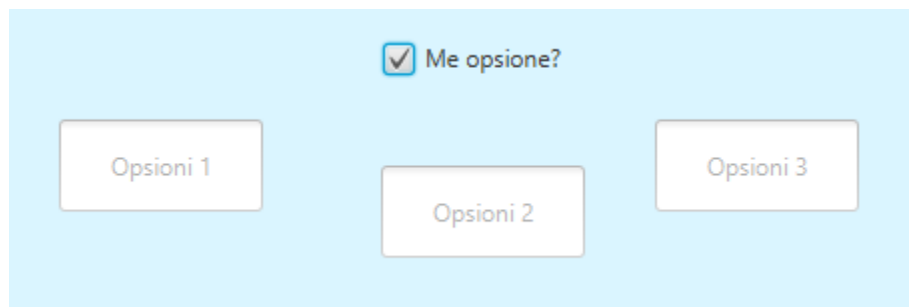
Siq edhe mund të supozohet nga emri, butoni vendosi pyetjet është buton i cili i lejon userit vendosjen e pyetjeve në database. Me shtypjen e këtij butoni ajo që afishohet te useri është paraqitur në figuren 2.2.



The screenshot shows a light blue rectangular area containing a form. At the top, there are two labels: "Shëno pyetjen:" and "Shëno Pikët". Below each label is a white rectangular input field. In the center, there is a checkbox with the text "Me opsione?". Below the checkbox is a yellow rectangular button labeled "Submit". At the bottom of the light blue area is a red rectangular button labeled "Fshij një pyetje". Below the entire light blue area is a yellow rectangular bar with the text "Go Back" centered in it.

Figura 2.2:Brenda butonit “Vendosi pyetjet”

Në këtë dritare shihen që janë fillimisht dy hapsira për të vendosur tekst, ku shenimet mbi këto hapsira japin udhëzimet se çfarë duhet të vendoset brenda tyre. Ndërsa checkboxi poshtë pyet userin nëse pyetja të cilën dëshiron ta vendos është me opsione ashtu që nëse pyetja është me opsione përdoruesit i ofrohet një mjedis komod për vendosjen e opsioneve. Me shtypjen e këtij checkboxi në dritaren e treguar në figuren 2.2, shtohet edhe pamja e treguar në figuren 2.3



The screenshot shows a light blue rectangular area. At the top, there is a checkbox with a blue checkmark and the text "Me opsione?". Below the checkbox, there are three white rectangular boxes arranged horizontally. The first box on the left is labeled "Opsioni 1", the middle box is labeled "Opsioni 2", and the box on the right is labeled "Opsioni 3".

Figura 2.3: Checkboxi “Me opsione”

Me apo pa opzione lloji i pyetjes, butoni “Submit” ka pozicion të pandjeshëm në dritare dhe ka qëllim ruajtjen e pyetjes në databazë, shfaqjen e një mesazhi gabimi nëse imputet nuk janë të vlefshme nga ana logjike dhe kthimin e kësaj dritareje në gjendjen fillestare. Pashtë këti butoni ndodhet butoni tjetër i hijezuar me të kuqe që mbanë shënimin “Fshij një pyetje” i cili shfaq një hapsirë ku mund të shënohet një numër dhe anash tij ndodhet butoni “Submit” i cili fshinë pyetjen në numrin e shkruar në hapsirën e tekstit. Figura 2.4, paraqet aksionin e këti butoni.

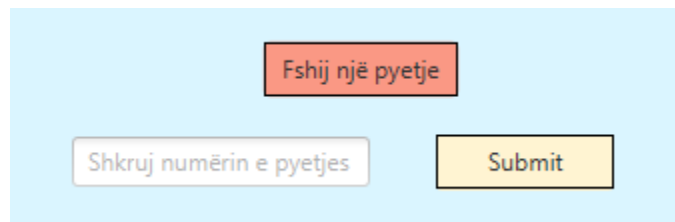


Figura 2.4: Butoni “Fshij një pyetje”

Ndërsa kërjt në fund të kësaj dritareje si dhe gjithë dritareve tjera pos dritares së parë është i vendosur butoni “Go Back”, buton i cili na kthen në dritaren komanduese.

2.2 Butoni “Gjenero Testin”

Butoni “Gjenero testin” është buton tjetër i cili shihet në dritaren kryesore dhe me klikimin e këtij butoni kalojmë në dritaren ku është i gjeneruar testi i cili mbanë pyetjet konform rregullit që shuma e poenave të atyre pyetjeve është 100. Mënyra se si bëhet përzgjedhja e këtyre pyetjeve do të sqarohet më poshtë tek arkitektura e aplikacionit. Figura 2.5, paraqet një gjenerimin e testit për disa pyetje

	Gjenero Testin	
<p>1)Cka quajme polimorfizem?? 30.0(piket)</p> <p>2)Nje klase mund te trashegoj sa klasa tjera? 25.0(piket)</p> <p>a)1 b)2 c)Pa kufi</p> <p>3)Per qka perdoret keyword extends? 45.0(piket)</p>		
Go back		

Figura 2.5:Butoni Gjenerimi i testit

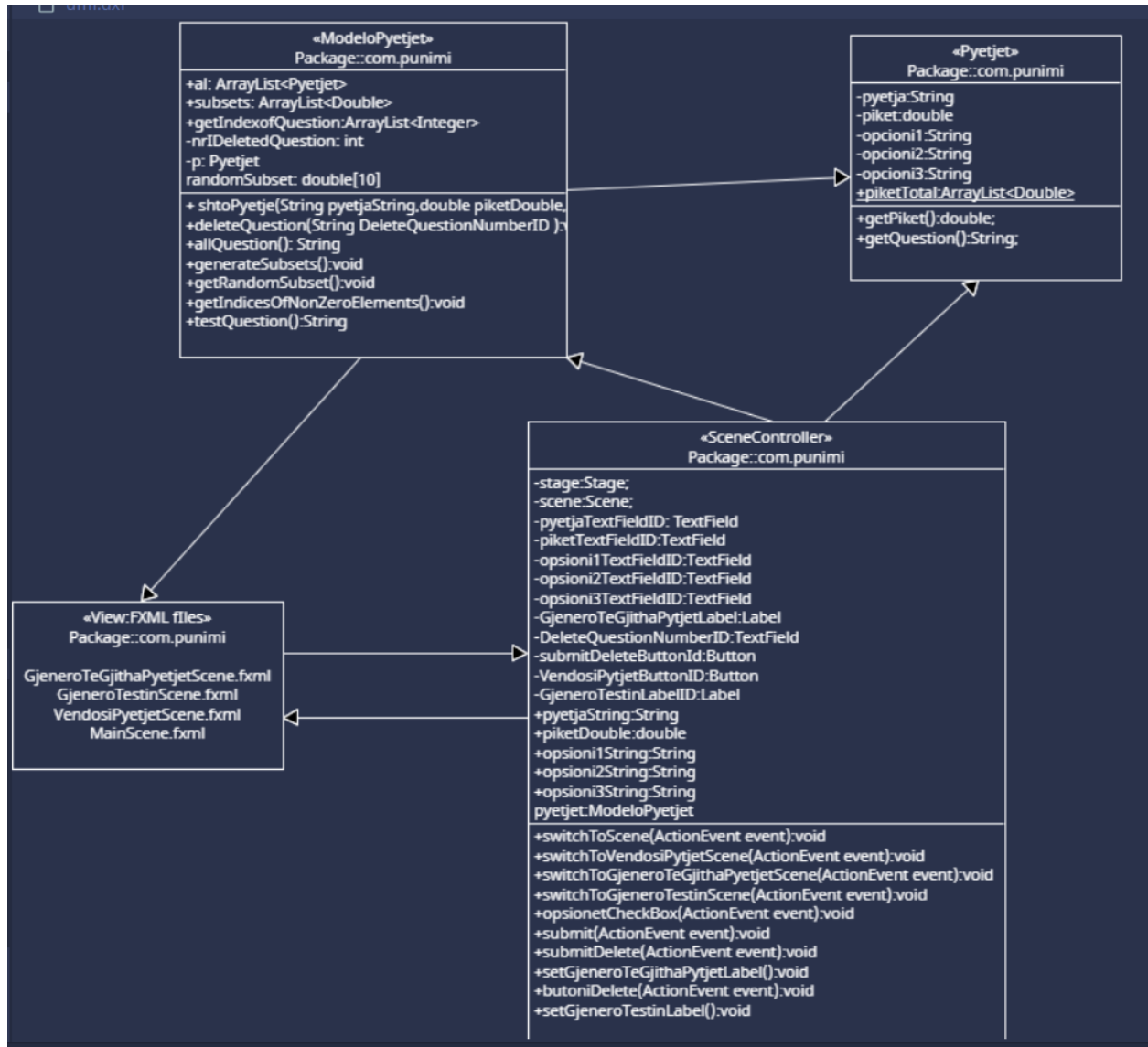
Butoni i fundit në dritaren kryesore është butoni i cili gjeneron të gjitha pyetjet që gjenden në bazën e të dhënave. Këtë buton e kemi vendosur për arsye se përdoruesi do të ketë mundësi të shoh nëse një pyetje e caktuar është ose jo në databaze, si dhe meqë modifikimi i pyetjeve bëhet duke ju referencuar me numra, atëherë përdoruesi duhet të dij numrin për pyetjen që dëshiron ta fshijë. Figura 2.6, paraqet veprimin e këtij butoni.



Fiugura 2.6:Butoni Gjenero te gjitha pyetjet

3.Arkitectura e Aplikacionit

3.1 Arkitektura e Aplikacion



3.1.Dokumentimi i kodit

Ndërtimin e ndërfaqes së aplikacionit me të cilin ndërvepron useri e kemi bere duke perdorur veglën SceneBuilder e cila ofron një mjedis me komod dhe na jep mundësi më të shpejta të ndërtimit të layoutit per javaFx.Meqenëse kemi përdoruar katër azhurnime te ndryshme per elementet e UI,duke perdorur SceneBuilder kemi modeluar 4 nderfaqe te ndryshme.Keto modelime i kemi marrur në katër fajlla te ndryshem fxml,te cilët pastaj i kemi perdorur ne kod si

shabllone te gatshme.Ne fajllat fxml ndodhen elementet e UI me te gjitha atributet e tyre si id-ja qe identifikon në mënyre unike elementin,ndodhet pozicioni i saktë për secilin prej elementeve qe do ta kete ne panele,ndodhet teksit te cilat ato elemente permbajne brenda vetes,per shume elemente te cilat shkaktojne ngjarje,ndodhet emri i metodes qe jep implementimin e duhur te tyre etj.

File fxml behet i perdorshem duke caktuar nje reference te tipit Parent(detyra e te cilit eshte te mbaje te grumbulluara elementet e UI) tek ky fajl,dhe kjo mundesohet permes klases FXXMLLoader e cila mabane nje metode statike te quajtur load.Shenimi i i meposhtem,paraqet nje pjese te kodit ku eshte marrur referenca e fajlit MainScene te tipit fxml.

```
Parent root=FXMLLoader.load(getClass().getResource("MainScene.fxml"));
```

Pastaj aplikohet forma standarte futja e variables se Parent ne Scene,Scena ne Stage dhe Stage behet e dukshme.

Klasa ScenController eshte klase e cila mbane rrejdhen e kontrollit te aplikacionit, keshtu qe ne kete klase permes metodave jane te trajtuara te gjitha ngjarjet e shkaktura nga elementet e UI.Ne fillimin e kesaj klase shohim nje numer te konsiderueshem te teksteve te cilat kane kete ndertim

```
@FXML private LLojiElementit IdElementit
```

Keto linja deklarojne variabla te fushes private te cilat asociojne me llojin e elementit ne fajllin fxml kurse shënimi @FXML përdoret për të shënuar fushën si të lidhur me një nyje specifike në skedarin FXML.

Metodat brenda kesaj klase shihet se shumica kan si parameter formal nje variabel te tipit “ActionEvent”,kjo sepse te gjitha keto metoda perdoren per trajtimin e ngjarjeve te shkaktuara nga elementet e UI,dhe pikerisht kjo klase(ActionEvent) trajton keto qeshtje.

Perbrenda te gjitha metodave te kesaj klase me ane te se cilave kerkohet nje azhurnim i ri i elementeve,inicializimi i Stage behet ne kete menyre

```
stage=(Stage)((Node)event.getSource()).getScene().getTioë();
```

Dhe ky kod mundeson qe te perdoret e njejta kornize per rregullimin e elementeve sa here qe thirret metoda qe ben icializimin e stage ne kete menyre.

Po ashtu ne ballinat e atyre metodave te cilat kerkojne imput nga perdoruesi verehet “throw IOException”permes kesaj deklarohen perjashtimet per impute jo te denja te cilat eventualisht mund te vendosen nga useri.

Sic edhe u cek me larte veprimet te cilat mundeson aplikacioni yne jane:perdoruesit i lejohet vendosja e numrit te pakufizuar te pyetjeve ne bazen e te dhenave,i lejohet fshirja e pyetjeve bazuar ne numrin e pyetjes,aplikacioni mundeson gjenerimin e testit qe ploteson kushtin qe poenat totale per pyetje te jene 100 si dhe perdorusi njehereshi mund te shoh te gjitha pyetjet te cilat i ka te vendosura ne bazen e te dhenave.

Te gjitha keto veprime te cilat u ceken me larte,jane te permbledhura ne klasen modeluese ModeloPyetjet.Struktura e perdorur per ruajtjen e pyetjeve te vendosura nga useri eshte ArrayList dhe kjo eshte edhe arsyeja e cila i lejon perdoruesit te vendos numer te pakufizuar te pyetjeve.

Konstruktori ModelPyetjet eshte perdorur vetem per inicializimin e variablave te fushes,gjithashtu ne kete klase mbajme referencen per tek klase tjeter Pyetjet,e cila sherben per azhurnimin e pyetjeve ne dritare,apo ne konteksin e MVC kjo klase luan rolin e output Vieë.

Per arsye se pyetjet kan nje numer te madh te vetive ruhen si objekte ne ArrayList,ndersa metoda e cila mundeson vendosjen e pyetjeve ketu eshte

```
shtoPyetje(String pyetja,double piket,String opcioni1,String opcioni2,String opcioni3)
```

Keta parametra formal per kete metode mirren nga klase SceneController konkretisht mirren nga teksti i cili vendoset ne textField dhe permes metodes add shtohen ne arrayList si objekte te tipit Pyetjet.

Metoda deleteQuestion(String numriPyetjes),thirret nga butoni Submit,kjo metode merr parameter te tipit String,ku permbajtja e ketij stringu duhet te jete nje numer pas kesaj stringu i dhene konvertohet ne variable te tipit int.Dhe ky numer perdoret per ta hequr nga arrayLista pyetjen e vleres se variable minus nje.

Metoda allQuestion() inicializohet nga metoda setGjeneroTeGjithaPyetjetLabel(). Perdore nje for loop per te kaluar ne cdo elemet te arrayListes,thirr metoden getQuestion() nga klase Pyetjet per cdo element te arrayListes, kjo metode jep nje String standart per cdo objekt. Keto Stringje grupohen se bashku ne nje variabel po ashtu qe tipit String,kjo variabel kthehet te metoda setGjeneroTeGjithaPyetjetLabel(). Dhe ky String behet i dukshem ne nje label,qe gjendet brenda kesaj metode.

Ne rastin kur useri shtyp butonin “Gjenero testin” qe ndodhet ne nderfaqen e pare,thirret metoda setGjeneroTestinLabel(),brenda asaj metoda ne nje variable te tipit String ruhet vlera e cila kthehet nga metoda testQuestion() e klases ModeloPyetjet.

Veprimet te cilat kryhen brenda metodes testQuestion() jane:

Fillimisht brenda keaj metode thirret metoda generateSubsets() e po kesaj klase,ku detyra e kesaj metode eshte gjenerimi i te gjitha nenbashkesive te poenave te pikeve,dhe bashkesite te cilat plotesojne kushtin qe shuma e elementeve te atyre bashkesive te jete 100 ruhen ne ArrayList(variabel e fushes) qe bane objekte te tipit double

Pastaj ne metoden testQuestion() thirret metoda tjeter getRandomSubset() e cila zgjedh nje nga arrayt nga AaryLista e gjeneruar nga metoda e mesiperme.

Pastaj thirret metoda getIndecesOfNonZeroElemnts().Duke pas parasysh qe Array qe na eshte gjenruar(ku shuma e elementeve eshte 100) permabe elementet te rradhitura ku indeksi i elementit qe permbane vlere te ndyshme nga zero perputhet me pytjen.Kjo metode naj jep indeksat e atyre elemnteve,keshtu qe ne baze te atyre indeksave ne dijmë tani se per cilen pyetje po behet fjale apo si tersi cilat pyetje formojne shumen e poenave 100.Ndersa ekzekutimi i pjeses tjeter te metodes paraqet shfaqjen e elementeve,qe eshte e njejte siq u tregua tek metoda allQuestion()

4.Rëndësia e aplikacionit.

Rëndësia e këtij aplikacioni është mjaft e madhe në kursimin e kohës për mësimdhënësit të cilët përpilojnë teste vazhimisht,për arsye se nuk kan nevoj që cdo herë të mirren me përpilimin e pyetjeve por vetëm me gjenerimin e tyre,po ashtu ndryshe nga programet për procesimin e tekstit që përdoren në krijimin e testeve kjo vegël duke larguar brengosjen e perdoruesit nga dizajni ofron nje mjedis me te pershtatshem per perpilim.Gjithhashtu vetia e ketij aplikacioni qe te gjeneroje nje total te pikeve fiks prej 100 bene qe pyetjet te kene pesha standarte rrjedhimisht testet te jene me ideale ndryshe nga perpilimi i cili behet me dore ku nese shuma e poenave nuk plotesone kushtin e 100 pikeve behen kompromise ne peshat e pikeve.

