UČILICA - PROGRAMERSKA DOKUMENTACIJA

Izradili : Ivona Čižić, Luka Jandrijević, Helena Marciuš, Dora Raštegorac

Link na github repozitorij: <https://github.com/jandra1/ucilica>

Cijeli projekt je izrađen koristeći C# programski jezik, uz pomoć windows formi i SQL bazi podataka. Za pokretanje aplikacije je potrebno instalirati visual studio, download link se može pronaći na sljedećem linku: <https://visualstudio.microsoft.com/> (potrebno je skinuti .NET framework od ponuđenih paketa) i softver je besplatan. Da bi vam radilo spajanje na bazu podataka, možda ćete morati skinuti i instalirati program sa sljedeće stranice: <https://www.microsoft.com/en-us/download/confirmation.aspx?id=13255> . Ukoliko još imate problema sa spajanjem na bazu podataka pri paljenju aplikacije, unutar “properties” od projekta, maknite kvačicu za “prefer 32 bit”.

U datoteci “pitanja i slike” nalaze se pitanja i pripadajuće slike koje idu uz pitanja koja se nalaze u učilici. Unutra su pitanja i odgovori koji će se prikazivati u aplikaciji. Pitanja su za 4.,5., 6., 7. i 8. razred osnovne škole te iz predmeta hrvatskog jezika, geografije, povijesti, matematike te engleskog jezika.

U opis-projekta.docx se može pronaći malo detaljniji opis kako bi aplikacija trebala raditi i čemu služi. U datoteci Ucilica se pronalazi Ucilica.sln koji se pokreće na početku za početak rada aplikacije.

**Ucilica/Ucilica**

U datoteci Ucilica/Ucilica nalaze se svi potrebni file-ovi potreban za rad aplikacije. Properties je datoteka koja se automatski generirala. U Resources možemo pronaći sve potrebne slike koje se koriste te audio zapise koji se puštaju tijekom korištenja aplikacije. Sve nove slike, audio zapise i slično, potrebno je tu staviti.

Svi file-ovi koji imaju nastavak Designer.cs ili .resx su fileovi koji se generiraju kada se na formu dodavaju gumbovi/polja ili ostale komponente te se generiraju od strane visual studija.

Početna forma je klasa Form1.cs u kojoj imamo string name u kojem čuvamo podatke za username pri logiranju u aplikaciju. Također, u njoj se spajamo na bazu podataka koja čuva username i password. Prilikom klika na gumb login poziva se funkcija Login\_Click koja obrađuje korisnikov unos te zahtjeva da upiše username i lozinku koja može sadržavati samo brojeve. Nakon toga, funkcija se pokušava spojiti s bazom podataka koja sadrži login podatke o pojedinom korisniku. Ako postoji točno jedna osoba u bazi podataka koja zadovoljava uvjete da se podudaravaju username i lozinka tada zatvaramo vezu s bazom i prelazimo na glavnu stranicu. Ako se prilikom ulogiravanja admin prijavio sa usernameom admin i lozinkom 1111 tada se prelazi na drugu formu.

Ako korisnik klikne na gumb za zatvaranje aplikacije, tada se forma gasi, a ako klikne na registraciju, otvara se posebna forma za registraciju.

Admin.cs je klasa od Forme pomoću koje korisnik ulazi u formu Dodaj.

Dodaj.cs je forma kojom admin dodaje nova pitanja u bazu. U klasi se nalazi lista predmeti koja sadrži sve predmete koji se nalaze na aplikaciji: geografija, matematika, hrvatski i povijest te kada se klikne na gumb, spajamo se na bazu podataka i dodajemo pitanje, odgovore i razred koji su upisani od strane admina u odgovarajuće textbox-ove.

U klasi Kraj, definiramo formu koja prikazuje vrijeme i finalni broj bodova. Također, klikom na gumb se forma zatvara.

Klasa Bodovi.cs nam služi za prikazivanje bodova svih usera koji su postigli rezultat igrajući učilicu. Ako u Combo box-u od predmeta ili razreda nije prazan tekst, tada pozivamo funkciju changeTable koja se brine o ispravnom prikazu svih rezultata na formi.

Pitanje.cs je dio forme koji sadržava varijable odgovorne za prikazivanje pitanja iz baze podataka. U konstruktoru inicijaliziramo razred, predmet te zvukove yay i wrong. Također pozivamo funkciju sljedećePitanje() koje prikazuje iduće pitanje iz baze podataka.

Postoje 4 moguća odgovora, pa će tako postojati i 4 funkcije koje se pozivaju na klik te svaka od njih provjerava je li kliknut odgovor točan, ako je svira određena melodija i broj bodova se povećava. U protivnom se gumb zacrveni i svira zvuk za krivi odgovor te se prikazuje točan odgovor tako što se pripradni gumb zazeleni.

Funkcija sljedećePitanje() je odgovorna za odabir sljedećeg pitanja iz baze podataka. Svaki kviz sadrži 9 pitanja pa ako je brojač pitanja veći od 9, onda se ne radi ništa. Prikazujemo stanje bodova na svakom idućem pitanju te pomoću randomizacije odabiremo iduće pitanje iz baze. Ako pitanje sadrži i sliku, tada prikazujemo i to u formi. Također prati na kojem smo pitanju.

U funkcijama initializeTime() i t\_Tick() brinemo o otkucaju sata te tako mjerimo vrijeme u sekundama i zapisujemo ga.

onemoguciGumbe/omoguciGumbe postavlja gumbe za odabir odgovora na false/true dok u funkciji hintButton\_Click prikrivamo pogrešno pitanje kada korisnik pritisne tipku za hint. U toj funkciji prvo gledamo koji odgovor je krivi, te taj odgovor postavljamo na crnu boju i postavljamo atribut tog gumba da se ne može kliknuti na njega.

Klasa Razred.cs, spajamo se na bazu podataka te čuvamo listu svih predmeta. Njih zatim prikazujemo u ComboBox-u. Klikom na gumb, prikazujemo formu pitanje i to tako da joj prosljeđujemo odabran predmet kako bi kasnije znali iz kojeg predmeta izvlačimo pitanja.

Klikom na drugi gumb zatvara se forma i otvara se forma za login.

U Register.cs klasi brinemo se o registraciji novih korisnika. Spajamo se na početku na bazu, te postavljamo veličinu forme u konstruktoru. Klikom na gumb za registraciju, prvo provjeravamo unos od strane korisnika. Polje za username i lozinku nesmije biti prazno te lozinka mora sadržavati minimalno 4 broja, bez slova. Zatim se pokušavamo spojit na bazu i u bazi pretražujemo je li zadani username koji je korisnik upisao već postoji. Ako postoji, obavješćujemo korisnika i ne radimo ništa. Također to radimo i za lozinku. Ako je sve ostalo u redu, tada dodajemo novog korisnika u bazu podataka.

Klasa database.cs služi za izvlačenje pitanja iz baze. Ona ima funkciju getQuestionsByYearAndSubject koja pomoću baze podataka dohvaća pitanja iz određenog razreda i predmeta. Izvuče sve moguće odgovore, slike i pitanja. Funkcija addQuestion služi za dodavanje pitanja u bazu podataka pomoću SQL naredbi. Također, postoji funkcija getScoresBySubjectAndYear u kojoj dohvaćamo iz baze podataka top 10 korisnika s najvećim brojem bodova za određeni razred i predmet. U ovoj klasi se brinemo i za upis novog broja bodova u bazu, pa za to postoji funkcija insertNewScore u kojoj u tablicu bodovi dodajemo novog korisnika s određenim brojem bodova i vremenom. Funkcija getResultsByName dohvaća sve rezultate određenog korisnika. Funkcije changeUserName, changePass i changeYear služe redoma za promjenu korisničkog imena, lozinke i razreda ordeđenog korisnika.

Klasa pitanjeKlasa.cs služi za spremanje i čuvanje pitanja i odgovore za potrebne predmete i razrede te za čuvanje informacija o slici i razredu na koje je pitanje namijenjeno. Ovu klasu koristimo na više mjesta.

Što treba popraviti ili dodati?

- više pitanja za ponuđene razrede

- dodati još razreda ili predmeta za učenje

- više pitanja uz sliku ili video

- kada je kviz pokrenut, moguće se logout-ati i nastaviti igrat kviz. Možda bi moglo staviti da kada se stisne logout za vrijeme kviza da se svi prozori zatvaraju.

- omogućiti kviz sa pitanjima iz više predmeta (korisnik ih sam bira)

- omogućti rješavanje kviza bez ulogiranja (tada se rezultat ne sprema u bazu)

- dodati gumb na početnoj formi u kojoj je moguće zvuk staviti na mute/unmute.

- dodati da svira neka pjesma tijekom igre.