# Dokumentacija o projektu iz objektno orijentisanog programiranja

## FrmLogIn:

Metode Save i LoadFromFile sluze za cuvanje i ucitavanje iz xml file-a. Pravim listu igraca van konstruktora, koristicu je kod Log In-a. U konstruktoru forme postavljam da se mesto teksta u textbox-u za password vide kruzici, kako se ne bi videla sifra ukoliko korisnik to zeli. Opciju da li da se password vidi ili ne resavam uz pomoc eventa check box-a, CheckChanged. U tom eventu proveravam da li je box checkovan ili ne i ukoliko jeste password mozemo videti kao text a ne kao bullet-e. Postoji takodje event btnSignUp\_Click kojim jednostavno otvaram formu za pravljenje naloga ukoliko korisnik nema svoj. Uz pomoc dve bool metode ProveraUsername i ProveraPassword, proveravam da li su username i password koje je korisnik uneo u textbox-ove za to ispravni. Ove dve metode pozivam u eventu btnLogIn\_Click, i ukoliko su ove dve metode true, korisnik se uspesno ulogovao na svoj account i otvara se forma FrmStartUp. Na ovoj formi postoje jos 4 eventa za klik i leave na text box-ovima za username i password. Sve rade na istom principu, ukoliko ne postoji tekst u njima onda ce biti ispisano ono sta treba da se unese, a ukoliko korisnik klikne na jedan od ovih text box-ova indicator ce nestati i korisnik ce slobodno moci da unosi sta zeli.

## FrmSignUp:

Metode Save i LoadFromFile su iste kao i u prethodnoj formi. U konstruktoru forme ucitavam igrace u listu igraca, kao i u FrmLogIn. Ista situacija sa text box-ovima, ukoliko nisu selectovani, pokazuju ono sto bi trebalo da se unese u njih. Check box-ovi rade takodje kao u FrmLogIn-u. Ovaj put postoje tri bool metode, pored ProvereUsername i ProveraPassword, postoji i ProveraConfirma. U ovoj formi prve dve metode proveravaju da li username i password koje je korisnik uneo u text box-ove ispunjavaju uslove, i da li vec mozda postoji isti username kao sto je korisnik uneo. ProveraConfirma jednostavno proverava da li su textovi u txtPass i txtConPass jednaki. Event btnCreate\_Click se izvrsava samo ako je rezultat prethodne tri metode true. Pravi nalog i cuva ga u xml file-u, i potom zatvara ovu formu.

## FrmStartUp:

Ova forma se prikazuje kad se korisnik prijavi na svoj nalog. Koristi se za prosledjivanje igraca koji se ulogovao do FrmMyDecks. U njoj postoje 4 buttona, i svaki ima svoj click event. btnQuit gasi program. btnSignOut ce se izlogovati iz naloga i otvoriti formu za log in. btnMyDecks otvara formu FrmMyDecks, a btnStart otvara web stranicu na kojoj se moze igrati YuGiOh.

## FrmMyDecks:

Cetiri metode se nalaze na pocetku, SaveCards, SaveDeck, LoadFromFileCards i LoadDeckFromFile, rade na principu metoda Save i LoadFromFile. Zatim postoji metoda Ispuni koja na pocetku ispuni listu svim kartama iz xml file-a. U konstruktoru forme load-ujem igrace i odredjujem koji je trenutno ulogovan, da bih mogao uspesno sacuvati deck samo za njega. Zatim u combo box-u ucitavam deckove od trenutnog korisnika ukoliko on ima neke sacuvane deckove. Na kraju ucitavam sve karte u listview. Koristim event tbSearch\_TextChanged za pretrazivanje karata. MouseClick eventovima nad listBox-ovima i nad listView-om omogucavam da se prikazuje informacija o trenutno izabranoj karti u picture box-u i rich text box-u, a DoubleMouseClick eventom omogucavam da se karta doda u deck, ili izbaci iz njega. Koristeci btnSave\_Click event cuvam sve sto se nalazi u trenutnim listBox-ovima za main i extra deck i cuvam to u jedan deck. Deck dobija ime po username-u korisnika + text u text box-u namenjenom za naziv decka. Potom refresujem combo box, da bi se prikazao i deck koji je sada sacuvan. Poslednji event koji koristimo ovde je cbDeckovi\_SelectedIndexChanged, koristimo ga kada korisnik odabere neki deck iz liste deckova koja se nalazi u combo box-u. Prikazuju se karte koje se nalaze u tom deck-u.

## Folder Duelist

U ovom folderu se nalaze klase pomocu kojim mi sve cuvamo u xml file-ovima. Tu su klase Igrac, koja predstavlja korisnika sa username-om, password-om i listom deck-ova. Klasa Deck sadrzi ime koje dobija po svom korisniku + neki unikatan naziv za deck koji mu korisnik dodeli. U nazivu se nalazi username korisnika jer je on unikatan i vazi samo za njega, i time samo on moze pristupiti svojim deck-ovima. Klasa Deck takodje ima dve liste Card-ova. Jedna lista predstavlja main deck, dok druga predstavlja extra deck. Klasa Card sadrzi property-e Name, Effect i dva enum tipa Place i TipK. Place predstavlja mesto gde karta treba da stoji, main ili extra deck, posto se karta ne moze nalaziti na oba mesta. TipK predstavlja da li je karta monster, spell, trap,…