

Strukturalni patern :

Iz ove grupe paterna iskorišten je Proxy Pattern, konkretno za kontrolu i osiguravanje računa korisnika. Onemogućena je dalja manipulacija ukoliko su uneseni pogrešni podaci, ukoliko neki uslov nije ispunjen, te ukoliko korisnik nema prava pristupa.

Kreacijski patern :

Iz ove grupe paterna iskorišten je Singleton Pattern, a koristi se u situacijama kada je potrebno imati samo jednu instancu neke klase. Uključuje samo jednu klasu koja je odgovorna za svoje instanciranje, kako bi bila sigurna da ne stvara više od jedne instance, a istovremeno pruža globalnu tačku pristupa toj instanci. Konkretno, patern je iskorišten za klasu SjediBa, koja predstavlja centralnu klasu programa.

Patern ponašanja :

Iz ove grupe paterna iskorišten je Observer Pattern, a cilj observer paterna je da definiše vezu jedan naprema više između objekata tako da kada se stanje jednog objekta promijeni, svi objekti koje to zanima dobivaju informaciju o toj promjeni. To predstavlja interfejs IObavijest, kojim se obavještavaju korisnici o nadolazećim događajima.