

Tokom projektovanja rješenja naše teme, došli smo na ideju da sve funkcionalnosti koje se tiču grafičkog interfejsa (navigacija kroz UI prozore i slično) realiziramo putem opcije Touchscreen na LCD displeju. Kao rezultat, dobili smo rješenje gdje je izbačena potreba za dodavanjem drugih komponenti kao što su potenciometar ili dugme.

U programu se također koriste slike predstavljene kao nizovi heksadecimalnih vrijednosti, od kojih svaka predstavlja jedan piksel, te koje su prikazivane na ekranu uz pomoć *built-in* metode `BSP_LCD_DrawPixel`.

Implementacija je sastavljena od dvije vrste metoda: metode koje su odgovorne za prikaz grafičkog interfejsa, te metode koje se tiču radne logike same igrice.

Što se tiče same logike kako igrice treba da se ponaša, u programu su implementirane sljedeće klase:

- `class Card`
- `class Deck`
- `class Blackjack`

Pored navedenih klasa, potrebne su i enumeracije:

```
enum Suits {  
    CLUB, HEART, SPADE, DIAMOND  
};  
  
enum Status {  
    Beginning, InProgress, Ended  
};
```

Klasa `Card` predstavlja jednu igračku kartu, te ima attribute: **value** – tipa `string` i **suit** – tipa `enum Suits` koja označava „familiju“ kojoj karta pripada. Pored standardnih *get* metoda i konstruktora, klasa ima funkciju koja je zadužena za prikazivanje karte na displeju:

```
void drawCard (int positionX, int positionY, bool hide = false)
```

Parametri:

- `positionX` – početna pozicija odakle počinje crtanje na X osi
- `positionY` – početna pozicija odakle počinje crtanje na Y osi
- `hide` – ako je `true` karta je okrenuta naopako tj. vidimo stražnju stranu karte

Ova metoda također poziva metode `drawSuit` i `drawNum` koje crtaju simbol karte te njen broj tj. slovo.

Klasa Deck sadrži niz karata (odnosno vektor) karata i ima ulogu špila. Posebno se ističe metoda **Deck shuffle ()** koja obavlja miješanje karata u špil u te vraća promiješani špil.

Klasa Blackjack je klasa sadrži najveću količinu logike. Pored get/set metoda, klasa sadrži i sljedeće metode koje implementiraju funkcionalnosti igrice:

void resetGame ();

Vrši reset svih atributa klase i postavlja atribut **gameStatus** na vrijednost **Beginning**.

int compareTotals ();

Poredi poene igrača i dilera te određuje ko je pobjednik runde.

void actionHit (int initialBet);

Implementira Hit funkcionalnost tj. dodaje novu kartu igraču te vrši provjeru da li je igrač prekoračio ili dostigao 21 poen.

void dealerPlays (bool natural, int initialBet);

Poziva se kada je red na dilera da odigra svoj potez.

void setUp (int initialBet);

Poziva se na početku svake runde blackjack-a.

Pored navedenih klasa i njihovih metoda, još jedna bitna metoda je:

void addCardsType (Suits suit, std::vector<Card> &cards);

Ova metoda je zadužena za dodavanje 13 karata u vektor **cards** za određenu familiju karata.

Drugi dio metoda odnosi se na grafički interfejs, te će u nastavku biti kratko opisane relevantne metode.

void drawEdge ();

Metoda zadužena za crtanje okvira većine prozora.

void drawMainMenu ();

Crta na displeju prozor glavnog menija.

void displaySetUp ();

Obavlja postavljanje boje, okvira i teksta „Back“ na prozoru.

void drawAbout ();

Zadužena za prikazivanje About prozora.

`void firstPage ();`

`void secondPage ();`

`void thirdPage ();`

Sve tri metode prikazuju tri stranice How To Play opcije.

`void drawHowTo (int nekst);`

Metoda koja u zavisnosti od parametra prikazuje određenu stranicu.

`void exitScreen ();`

Crta prozor nakon klika na opciju Exit.

`int startScreen ();`

Metoda iscrtava prozor gdje korisnik bira početne žetone. Kao rezultat vraća broj izabranih žetona.

`void drawChip ();`

Crta ikonicu pored trenutnog broja žetona.

`void drawPlaceBet (int chips, int score, int bet);`

Iscrtava prozor nakon klika na opciju Place bet.

Parametri:

- chips – broj žetona koji će se ispisati na ekranu.
- score – postignuti poeni igrača.
- bet – vrijednost trenutne opklade koja će biti ispisana na ekranu.

`void drawScore (int score, bool bottomedOut = false);`

Iscrtava prozor koji prikazuje krajnji rezultat igrača.

Parametri:

- score – postignuti poeni igrača.
- bottomedOut – ako parametar ima vrijednost true, ispisuje se dodatna poruka na ekranu.

`void drawPlay (Blackjack blackjack);`

Iscrtava prozor gdje se odvija igra.

Parametri:

- blackjack – trenutna runda

`void prepCard ();`

Priprema osnovni oblik karte (bijeli okvir karte na koji se naknadno postavlja simbol i broj).

`void drawHighScore ();`

Iscrtava prozor nakon klika na opciju Highscore.

U main funkciji obavlja se inicijalizacija touchscreen-a, te priprema karata i kreiranje objekta klase Blackjack. Unutar while petlje se nalazi jedan oblik listenera koji detektuje dodir te na osnovu koordinata x1 i y1 određuje koja je opcija kliknuta, te poziva odgovarajuće UI metode i metode Blackjack klase.