ПРИЛОЖЕНИЕ Б

УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ
«БРЕСТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
Кафедра Интеллектуальных информационных технологий

РАЗРАБОТКА ЭЛЕКТРОННОЙ ККИ

ТЕКСТ ПРОГРАММЫ

КР.ПО4.190333-05 12 00
Листов 11

Руководитель	А. А. Крощенко
Выполнил	П. И. Галанин
Консультант по ЕСПД	А. А. Крощенко

2022

Исходный код клиентской части

- **index.js** - входной файл WebPack
- **App.js** - главный компонент ReactJS
- **Menu.jsx** - страница с меню
- **ResultsTable.jsx** - страница со статистикой игр
- **GameController.js** - логика игры
- **GameView.jsx** - логика игры и её отображение

Листинг: ../sources/browser_game/src/index.js

```
import React from 'react';
import { createRoot } from 'react-dom/client';
import { BrowserRouter } from 'react-router-dom';

import './index.css';
import App from './App';

const container = document.getElementById('root');
const root = createRoot(container);
root.render(
  <BrowserRouter>
    <App />
  </BrowserRouter>,
);
```

Листинг: ../sources/browser_game/src/App.js

```
import { Routes, Route } from 'react-router-dom';

import Menu from './components/Menu/Menu';
import GameView from './components/Game/GameView';
import ResultsTable from './components/ResultsTable/ResultsTable';

function App() {
  return (
    <Routes>
      <Route path="/singleplayer" element={<GameView />} />
      <Route path="/statistic" element={<ResultsTable />} />
      <Route path="/" element={<Menu />} />
      <Route path="*" element={<Menu />} />
    </Routes>
  );
}

export default App;
```

Листинг: ../sources/browser_game/src/components/Menu/Menu.jsx

```
import styles from './Menu.module.css';
import { Link } from 'react-router-dom';

function Menu() {
  return (
    <div className={styles.wrapper}>
      <ul>
        <li>
          <Link to="/singleplayer">Одиничная игра</Link>
        </li>
        <li>
          <Link to="/statistic">Статистика игр</Link>
        </li>
      </ul>
    </div>
  );
}

export default Menu;
```

Листинг: ../sources/browser_game/src/components/ResultsTable/ResultsTable.jsx

```
import axios from 'axios';
import { useEffect, useState } from 'react';

import styles from './ResultsTable.module.css';

function ResultsTable() {
  const [array_data, set_array_data] = useState([{}]);

  useEffect(function () {
    (async function () {
      await get_statistic_data();
    })();

    if (array_data.length === 0) {
      alert('Нет данных в статистике');
    }
  }, []);

  async function get_statistic_data() {
    try {
      const url = 'http://localhost:3001/api/GameResults';
      const response = await axios.get(url);
      const array = response.data.data;
      console.log(array);
      set_array_data(array);
    } catch (err) {
      console.log(err);
    }
  }

  return (
    <table className={styles.table}>
      <thead>
        <tr>
          <td>№</td>
          <td>Player1 name</td>
          <td>Player2 name</td>
          <td>Player1 damage</td>
          <td>Player2 damage</td>
        </tr>
      </thead>
      <tbody>
        {array_data.map((element, index) => {
          return (
            <tr key={index}>
              <td>{element.code}</td>
              <td>{element.player1}</td>
              <td>{element.player2}</td>
              <td>{element.player1_damage}</td>
              <td>{element.player2_damage}</td>
            </tr>
          );
        })}
      </tbody>
    </table>
  );
}

export default ResultsTable;
```

Листинг: ../sources/browser_game/src/components/Game/GameController.js

```
import archer_4 from './archer_card_4.png';
import archer_5 from './archer_card_5.png';
import archer_6 from './archer_card_6.png';
import artillery_1 from './artillery_card_1.png';
import artillery_5 from './artillery_card_5.png';
import artillery_6 from './artillery_card_6.png';
import artillery_8 from './artillery_card_8.png';
import swordsman_1 from './swordsman_card_1.png';
import swordsman_2 from './swordsman_card_2.png';
import swordsman_4 from './swordsman_card_4.png';
import swordsman_5 from './swordsman_card_5.png';

let cards = [
  {
    id: 1,
    type: 'archer',
    damage: 4,
    img: archer_4,
    where: 'onHome',
  },
  {
    id: 2,
    type: 'archer',
    damage: 5,
    img: archer_5,
    where: 'onHome',
  },
  {
    id: 3,
    type: 'archer',
    damage: 6,
    img: archer_6,
    where: 'onHome',
  },
  {
    id: 4,
    type: 'artillery',
    damage: 1,
    img: artillery_1,
    where: 'onHome',
  },
  {
    id: 5,
    type: 'artillery',
    damage: 5,
    img: artillery_5,
    where: 'onHome',
  },
  {
    id: 6,
    type: 'artillery',
    damage: 6,
    img: artillery_6,
    where: 'onHome',
  },
  {
    id: 7,
    type: 'artillery',
    damage: 8,
    img: artillery_8,
    where: 'onHome',
  },
  {
    id: 8,
    type: 'swordsman',
    damage: 1,
    img: swordsman_1,
    where: 'onHome',
  },
  {
    id: 9,
    type: 'swordsman',
    damage: 2,
    img: swordsman_2,
    where: 'onHome',
  },
  {
    id: 10,
    type: 'swordsman',
    damage: 4,
    img: swordsman_4,
    where: 'onHome',
  },
  {
    id: 11,
    type: 'swordsman',
    damage: 5,
    img: swordsman_5,
    where: 'onHome',
  },
];

class Card {
  constructor(id, type, damage, img, where) {
    this.id = id;
    this.type = type;
    this.damage = damage;
    this.img = img;
    this.where = where;
  }

  take_card() {
    this.where = 'onWar';
  }
}

class GameController {
  constructor() {
    this.player1_cards = [];
    for (let i = 0; i < cards.length; ++i) {
      let id = cards[i].id;
      let type = cards[i].type;
      let damage = cards[i].damage;
      let img = cards[i].img;
      let where = cards[i].where;
      this.player1_cards.push(new Card(id, type, damage, img, where));
    }

    this.player2_cards = [];
    for (let i = 0; i < cards.length; ++i) {
      let id = cards[i].id;
      let type = cards[i].type;
      let damage = cards[i].damage;
      let img = cards[i].img;
      let where = cards[i].where;
      this.player2_cards.push(new Card(id, type, damage, img, where));
    }
  }

  player1_take_card(id) {
    for (let i = 0; i < this.player1_cards.length; ++i) {
      if (this.player1_cards[i].id === id) {
        this.player1_cards[i].where = 'onWar';
      }
    }
  }

  player2_take_card(id) {
    for (let i = 0; i < this.player2_cards.length; ++i) {
      if (this.player2_cards[i].id === id) {
        this.player2_cards[i].where = 'onWar';
      }
    }
  }

  get_player1_cards() {
    return this.player1_cards;
  }

  get_player2_cards() {
    return this.player2_cards;
  }

  get_player1_damage(type) {
    let count = 0;
    this.player1_cards.forEach((element) => {
      if (element.where === 'onWar' && element.type === type) {
        count += element.damage;
      }
    });
    return count;
  }

  get_player1_swordsman_damage() {
    return this.get_player1_damage('swordsman');
  }

  get_player1_archer_damage() {
    return this.get_player1_damage('archer');
  }

  get_player1_arlillery_damage() {
    return this.get_player1_damage('artillery');
  }
```

```
  get_player2_damage(type) {
    let count = 0;
    this.player2_cards.forEach((element) => {
      if (element.where === 'onWar' && element.type === type) {
        count += element.damage;
      }
    });
    return count;
  }

  get_player2_swordsman_damage() {
    return this.get_player2_damage('swordsman');
  }

  get_player2_archer_damage() {
    return this.get_player2_damage('archer');
  }

  get_player2_arlillery_damage() {
    return this.get_player2_damage('artillery');
  }

  get_player1_all_damage() {
    let count = 0;
    this.player1_cards.forEach((element) => {
      if (element.where === 'onWar') {
        count += element.damage;
      }
    });
    return count;
  }

  get_player2_all_damage() {
    let count = 0;
    this.player2_cards.forEach((element) => {
      if (element.where === 'onWar') {
        count += element.damage;
      }
    });
    return count;
  }
}

export default GameController;
```

Листинг: ../sources/browser_game/src/components/Game/GameView.jsx

```
import React, { useEffect, useState } from 'react';
import styles from './GameView.module.css';
import GameController from './GameController';
import EndGameWindow from './../EndGameWindow/EndGameWindow';

function GameView() {
  const [game_instance, setGame_instance] = useState(new GameController());
  const [player1_button_disable, set_player1_button_disable] = useState(false);
  const [player2_button_disable, set_player2_button_disable] = useState(true);
  const [player1_cards, set_player1_cards] = useState([]);
  const [player2_cards, set_player2_cards] = useState([]);

  const [player1_pas, set_player1_pas] = useState(false);
  const [player1_damage, set_player1_damage] = useState(0);
  const [player1_swordsman_damage, set_player1_swordsman_damage] = useState(0);
  const [player1_archer_damage, set_player1_archer_damage] = useState(0);
  const [player1_artillery_damage, set_player1_artillery_damage] = useState(0);
const [player2_pas, set_player2_pas] = useState(false);
const [player2_damage, set_player2_damage] = useState(0);
const [player2_swordsman_damage, set_player2_swordsman_damage] = useState(0);
const [player2_archer_damage, set_player2_archer_damage] = useState(0);
const [player2_artillery_damage, set_player2_artillery_damage] = useState(0);

useEffect(function () {
  set_player1_cards(game_instance.get_player1_cards());
  set_player2_cards(game_instance.get_player2_cards());
});

function player1_button_clicked(id) {
  if (player1_button_disable === true) {
    return;
  }

  if (player1_pas === true) {
    return;
  }

  if (player2_pas === false) {
    set_player1_button_disable(true);
    set_player2_button_disable(false);
  }

  game_instance.player1_take_card(id);

  set_player1_cards(game_instance.get_player1_cards());
  set_player1_damage(game_instance.get_player1_all_damage());
  set_player1_swordsman_damage(game_instance.get_player1_swordsman_damage());
  set_player1_archer_damage(game_instance.get_player1_archer_damage());
  set_player1_artillery_damage(game_instance.get_player1_arlillery_damage());
}

function player2_button_clicked(id) {
  if (player2_button_disable === true) {
    return;
  }

  if (player2_pas === true) {
    return;
  }

  if (player1_pas === false) {
    set_player2_button_disable(true);
    set_player1_button_disable(false);
  }

  game_instance.player2_take_card(id);

  set_player2_cards(game_instance.get_player2_cards());
  set_player2_damage(game_instance.get_player2_all_damage());
  set_player2_swordsman_damage(game_instance.get_player2_swordsman_damage());
  set_player2_archer_damage(game_instance.get_player2_archer_damage());
  set_player2_artillery_damage(game_instance.get_player2_arlillery_damage());
}

function player1_pas_clicked() {
  if (player1_pas === true) {
    return;
  }
  set_player1_pas(true);
  set_player1_button_disable(true);

  if (player2_pas === false) {
    set_player2_button_disable(false);
  }
}

function player2_pas_clicked() {
  if (player2_pas === true) {
    return;
  }

  set_player2_pas(true);
  set_player2_button_disable(true);

  if (player1_pas === false) {
    set_player1_button_disable(false);
  }
}

return (
  <div className={styles.game_wrapper}>
    {player1_pas === true && player2_pas === true ? (
      <EndGameWindow
        player1_name={'Player1'}
        player2_name={'Player2'}
        player1_damage={player1_damage}
        player2_damage={player2_damage}
      />
    ) : (
      <React.Fragment></React.Fragment>
    )}
    <div className={styles.player_s_card_field}>
      <div className={styles.statistic_block}></div>
      <div className={styles.game_cards_block}>
        {player2_cards.map((element, index) => {
          return element.where === 'onHome' ? (
            <img
              className={
                player2_button_disable
                  ? styles.image_disables
                  : styles.image_enabled
              }
              key={index}
              src={element.img}
              alt=""
              onClick={(event) => player2_button_clicked(element.id)}
            />
          ) : (
            <React.Fragment key={index}></React.Fragment>
          );
        })}
      </div>
      <div className={styles.statistic_block}></div>
    </div>
    <div className={styles.heavy_weapons_field}>
      <DamageBlock damage={player2_artillery_damage} />
      <div className={styles.game_cards_block}>
        {player2_cards.map((element, index) => {
          return element.where === 'onWar' && element.type === 'artillery' ? (
            <img key={index} src={element.img} alt="" />
          ) : (
        <React.Fragment key={index}></React.Fragment>
      );
    })}
  </div>
  <div className={styles.statistic_block}></div>
</div>
<div className={styles.field_of_archers}>
  <DamageBlock damage={player2_archer_damage} />
  <div className={styles.game_cards_block}>
    {player2_cards.map((element, index) => {
      return element.where === 'onWar' && element.type === 'archer' ? (
        <img key={index} src={element.img} alt="" />
      ) : (
        <React.Fragment key={index}></React.Fragment>
      );
    })}
  </div>
  <DamageBlock damage={player2_damage} />
</div>
<div className={styles.field_of_swordsmen}>
  <DamageBlock damage={player2_swordsman_damage} />
  <div className={styles.game_cards_block}>
    {player2_cards.map((element, index) => {
      return element.where === 'onWar' && element.type === 'swordsman' ? (
        <img key={index} src={element.img} alt="" />
      ) : (
        <React.Fragment key={index}></React.Fragment>
      );
    })}
  </div>
  <div className={styles.statistic_block}>
    <button
      className={`${
        player2_button_disable
          ? styles.button_disables
          : styles.button_enabled
      } ${player2_pas === true ? styles.button_pas : ''}`}
      onClick={(event) => player2_pas_clicked()}
    >
      ПАС
    </button>
  </div>
</div>
<div className={styles.field_of_swordsmen}>
  <DamageBlock damage={player1_swordsman_damage} />
  <div className={styles.game_cards_block}>
    {player1_cards.map((element, index) => {
      return element.where === 'onWar' && element.type === 'swordsman' ? (
        <img key={index} src={element.img} alt="" />
      ) : (
        <React.Fragment key={index}></React.Fragment>
      );
    })}
  </div>
  <div className={styles.statistic_block}>
    <button
      className={`${
        player1_button_disable
          ? styles.button_disables
          : styles.button_enabled
      } ${player1_pas === true ? styles.button_pas : ''}`}
      onClick={(event) => player1_pas_clicked()}
    >
```

```
            ПАС
          </button>
        </div>
      </div>
      <div className={styles.field_of_archers}>
        <DamageBlock damage={player1_archer_damage} />
        <div className={styles.game_cards_block}>
          {player1_cards.map((element, index) => {
            return element.where === 'onWar' && element.type === 'archer' ? (
              <img key={index} src={element.img} alt="" />
            ) : (
              <React.Fragment key={index}></React.Fragment>
            );
          })}
        </div>
        <DamageBlock damage={player1_damage} />
      </div>
      <div className={styles.heavy_weapons_field}>
        <DamageBlock damage={player1_artillery_damage} />
        <div className={styles.game_cards_block}>
          {player1_cards.map((element, index) => {
            return element.where === 'onWar' && element.type === 'artillery' ? (
              <img key={index} src={element.img} alt="" />
            ) : (
              <React.Fragment key={index}></React.Fragment>
            );
          })}
        </div>
        <div className={styles.statistic_block}></div>
      </div>
      <div className={styles.player_s_card_field}>
        <div className={styles.statistic_block}></div>
        <div className={styles.game_cards_block}>
          {player1_cards.map((element, index) => {
            return element.where === 'onHome' ? (
              <img
                className={
                  player1_button_disable
                    ? styles.image_disables
                    : styles.image_enabled
                }
                key={index}
                src={element.img}
                alt=""
                onClick={(event) => player1_button_clicked(element.id)}
              />
            ) : (
              <React.Fragment key={index}></React.Fragment>
            );
          })}
        </div>
        <div className={styles.statistic_block}></div>
      </div>
    </div>
  );
}

function DamageBlock(props) {
  return <div className={styles.statistic_block}>{props.damage}</div>;
}

export default GameView;
```