

Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования «Алтайский государственный технический
университет им. И.И. Ползунова»

Факультет информационных технологий
Кафедра "Прикладная математика"

Отчет защищен с оценкой _____

Преподаватель _____ А.А. Третьяков

« _____ » _____ 2018г.

ОТЧЕТ ПО РАСЧЕТНОМУ ЗАДАНИЮ

по дисциплине
«Современные средства промышленной разработки ПО»
РЗ 09.03.04.12.000 О

Студент группы ПИ-61

В.Р. Оверченко

Преподаватель доцент, к.ф.-м.н.

А.А. Третьяков

БАРНАУЛ 2020

Введение

В данном расчётном задании было реализовано приложение для анализа учебных материалов на совместимость на основе выделения ключевых слов данных текстов. Совместимость материалов рассчитывается в процентах.

Разработанное приложение позволяет просматривать список ключевых слов для каждого текста, удалять или добавлять слова в зависимости от их значимости на усмотрение пользователя. Пользователям приложения доступна регистрация в системе.

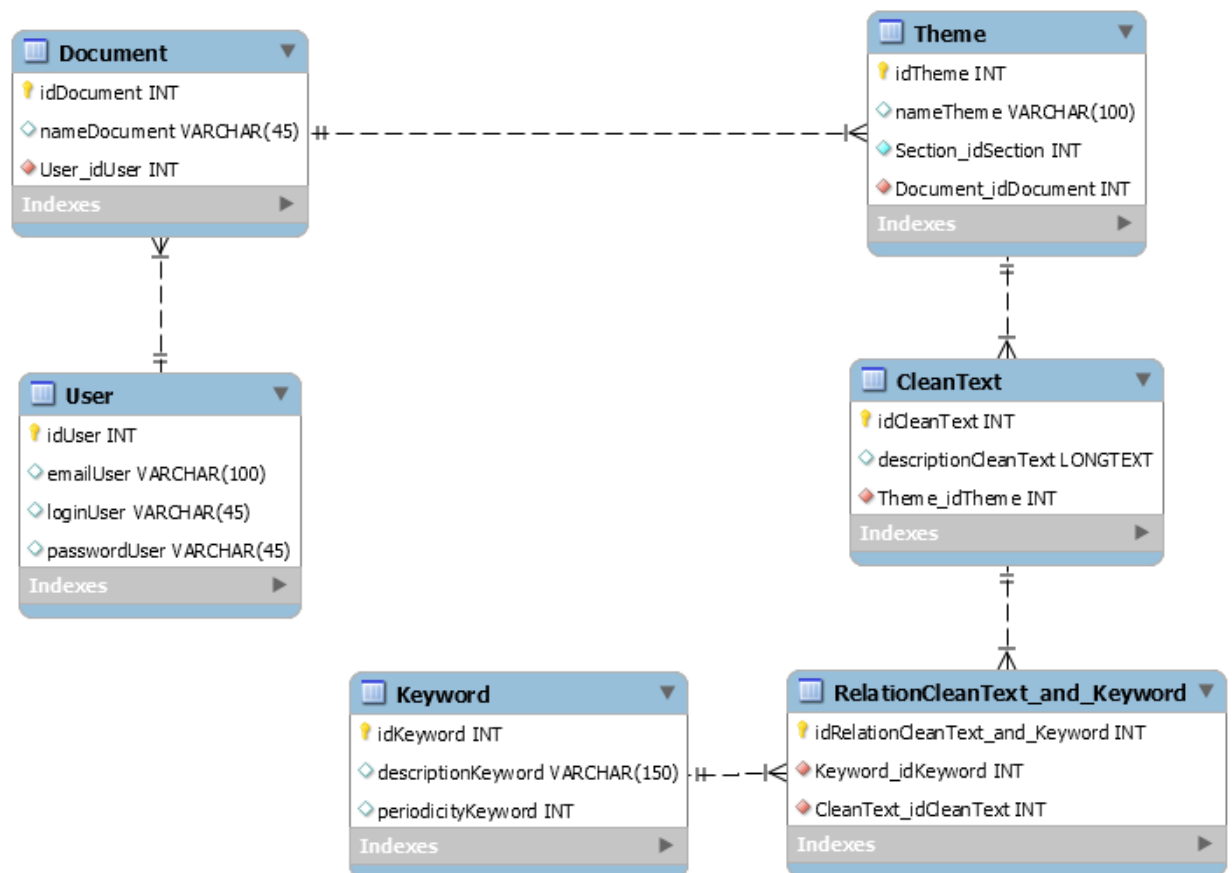
1. Постановка задачи

Имеется сервис, предназначенный для анализа учебных материалов на совместимость посредством выделения в текстах ключевых слов и их количественной оценки. С помощью сервиса можно просматривать все ключевые слова текста, редактировать, удалять и добавлять слова.

С приложением могут работать несколько видов пользователей. Зарегистрированные пользователи – пользователи, прошедшие регистрацию и авторизацию в приложении. Для таких пользователей сохраняется история их загрузок. Незарегистрированные пользователи – пользователи, не прошедшие регистрацию. Для них доступны данные в течение одной сессии.

2. Проектирование

2.1. Схема базы данных



3. Описание программного обеспечения

3.1. Описание использованных языка, фреймворка, СУБД

В качестве средств разработки были выбраны:

- 1) Node JS – программная платформа;
- 2) MySQL – СУБД;
- 3) Bootstrap 4 – CSS-фреймворк;
- 4) Express.js – фреймворк;
- 5) Handlebars – шаблонизатор.

Node или **Node.js** — программная платформа, основанная на движке V8 (транслирующем JavaScript в машинный код), превращающая JavaScript из узкоспециализированного языка в язык общего назначения. Node.js добавляет возможность JavaScript взаимодействовать с устройствами ввода-вывода через свой API, подключать другие внешние библиотеки, написанные на разных языках, обеспечивая вызовы к ним из JavaScript-кода.

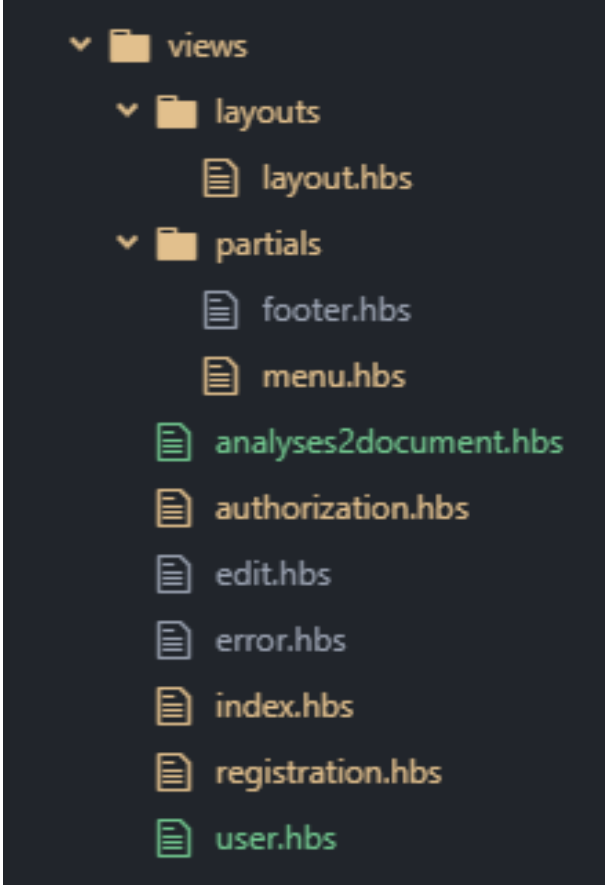
Express - это минималистичный и гибкий веб-фреймворк для приложений Node.js, предоставляющий обширный набор функций для мобильных и веб-приложений.

MySQL – реляционная СУБД. Имеет открытый исходный код. Является решением для малых и средних приложений. MySQL поддерживает SQL (структурированный язык запросов) и может применяться в качестве SQL-сервера. Плюсами MySQL являются: его простота, надежность и быстродействие.

Bootstrap 4 – это один из «стандартных» CSS-фреймворков. Использующихся для создания красивой и удобной адаптивной верстки. Прост в использовании.

Handlebars - это шаблонный процессор, который динамически генерирует HTML-страницу, что экономит время на ручном обновлении.

3.2. Views

	<p>Шаблон всех страниц</p> <p>Шаблон футера</p> <p>Шаблон меню</p> <p>Страница анализа 2 документов</p> <p>Страница авторизации пользователя</p> <p>Страница редактирования ключевых слов</p> <p>Страница ошибки</p> <p>Главная страница</p> <p>Страница регистрации пользователя</p> <p>Страница пользователя</p>
--	--

Заключение

В результате выполнения расчетной работы было разработано веб-приложение для анализа учебных материалов на совместимость по ключевым словам, обладающее следующими функциональными возможностями:

- Просмотр списка ключевых слов учебных материалов;
- Регистрация пользователей системы;
- Авторизация пользователей;
- Добавления, редактирование и удаление ключевых слов учебных материалов;

Данное веб-приложение имеет большой потенциал для расширения функциональности.

Список использованной литературы

1. Документация Bootstrap [Электронный ресурс]. – Режим доступа: <https://getbootstrap.com/docs/4.3/getting-started/introduction/>
2. Документация Express [Электронный ресурс]. – Режим доступа: <https://expressjs.com/en/guide/routing.html>
3. Документация NodeJS [Электронный ресурс]. – Режим доступа: <https://nodejs.org/en/docs/>
4. User login and registration with Node.js using Express, Bcrypt and MySQL [Электронный ресурс]. – Режим доступа: <https://medium.com/@siddarthasiddu96/user-login-and-registration-with-nodejs-using-express-bcrypt-and-mysql-529c872db5a0>
5. Creating Login and Registration Application Using NodeJS and MySQL [Электронный ресурс]. – Режим доступа: <https://coding-karma.com/2017/08/12/creating-login-registration-using-nodejs-mysql/>

Приложение А. Код приложения

Connection.js

```
var mysql = require('mysql');

var db;
var setting = {
  host : 'localhost',
  user : 'root',
  password : '',
  database : 'analysiseducationalmaterial'
};

function connectDatabase() {
  if(!db) {
    db = mysql.createConnection(setting);

    db.connect( function(err) {
      if (!err){
        console.log('Database connection!');
      }
      else {
        console.log('Error connection');
      }
    });
  }
  return db;
}

module.exports = connectDatabase();
```

Index.js

```
var express = require('express');
var router = express.Router();
var connection = require('../config/connection');

/* Основная страница (вывод всех слов) */
router.get('/', function(req, res) {
  connection.query('SELECT * FROM keyword', function(err, rows) {
    if (err) throw err;
    res.render('index', { words: rows });
  });
});

// Добавление слова
router.post('/addWord', function(req, res) {
  const wordData = {
```

```

        descriptionKeyword: req.body.descriptionKeyword,
        periodicityKeyword: 0
    }
    connection.query('INSERT INTO keyword SET ?', wordData, function(err, result){
        if (err) throw err;
        res.redirect('/');
    } )
});

// Удаление слова
router.get('/deleteWord/:idWord', function(req, res) {
    var wordid = req.params.idWord;
    connection.query('DELETE FROM keyword WHERE idKeyword = ?', [wordid], function(err,
rows){
        if (err) throw err;
        res.redirect('/');
    } )
})

// Переход на страницу редактирования слова
router.get('/edit/:idWord' , function(req, res) {
    var wordid = req.params.idWord;
    connection.query('SELECT * FROM keyword WHERE idKeyword = ?', [wordid], function(err,
rows){
        if (err) throw err;
        res.render('edit', { wordData: rows });
    } )
})

// Редактирование слова
router.post('/updateWord/:idWord', function(req, res) {
    var description = req.body.descriptionKeyword;
    var updateId = req.params.idWord;
    connection.query('UPDATE keyword SET descriptionKeyword=? WHERE idKeyword = ?',
[description, updateId], function(err, result){
        if (err) throw err;
        res.redirect('/');
    } )
});

// Переход на страницу регистрации
router.get('/registration', function(req, res) {
    res.render('registration');
});

// Переход на страницу регистрации

```

```

router.get('/authorization', function(req, res) {
    res.render('authorization');
});

// Переход на страницу анализа 2 текстов
router.get('/analyses2document', function(req, res) {
    res.render('analyses2document');
});

// Переход на страницу анализа 2 текстов
router.get('/user', function(req, res) {
    res.render('user');
});
/* Тест для ajax */
router.get('/testajax', function(req, res) {
    /*connection.query('SELECT * FROM keyword', function(err, rows) {
        if (err) throw err;
        res.render('index', { words: rows });
    });*/
    connection.query('SELECT * FROM keyword', function(err, rows) {
        if (err) throw err;
        var words = rows;
        res.send(words);
    });
});
module.exports = router;

```

Registration.js

```

const bcrypt = require('bcrypt');
var express=require("express");
var connection = require('../config/connection');
var router = express.Router();

router.post('/addUser', function(req, res) {
    if(req.body.password == req.body.passwordConfirm){
        const passwordHash = bcrypt.hashSync(req.body.password,10);
        var userData={
            "emailUser":req.body.email,
            "loginUser":req.body.email,
            "passwordUser":passwordHash
        }
        connection.query('INSERT INTO user SET ?', userData, function(err, result){
            if (err) throw err;
            res.redirect('/');
        });
    }
});

```

```

    } )
  }
  else {
    res.render('registration', {message: 'Пароли не совпадают!'});
  }
});

module.exports = router;

```

Authorization.js

```

const bcrypt = require('bcrypt');
var express=require("express");
var connection = require('../config/connection');
var router = express.Router();

router.post('/authorization', function(req, res){
  var email = req.body.email;
  var password = req.body.password;

  connection.query('SELECT * FROM user WHERE emailUser = ?', [email], function (err,
results, fields) {
    if (err) throw err;
    else{
      console.log(results);
      if(results.length >0){
        passwordHash = results[0].passwordUser;
        if(password == passwordHash){
          var id = results[0].idUser;
          console.log(id);
          res.redirect('/user');
          /*res.json({
            status:true,
            message:'successfully authenticated'
          })*//
        }else{
          res.json({
            status:false,
            message:"Email and password does not match"
          });
        }
      }
    }
  }
  else{
    res.json({

```

```

        status:false,
        message:"Email does not exists"
    });
    }
    }
    });
});
module.exports = router;

```

App.js

```

var createError = require('http-errors');
var express = require('express');
var path = require('path');
var cookieParser = require('cookie-parser');
var logger = require('morgan');
var bodyParser = require('body-parser');
var hbs = require('express-handlebars');

var indexRouter = require('./routes/index');
var registrationRouter = require('./routes/registration');
var authorizationRouter = require('./routes/authorization');

var app = express();

// view engine setup
app.engine('hbs', hbs({extname: 'hbs', defaultLayout:'layout', layoutsDir: __dirname +
'/views/layouts/'}));
app.set('view engine', 'hbs');
app.set('views', path.join(__dirname, 'views'));
app.use(express.static(__dirname + '/public'));

app.use(logger('dev'));
app.use(express.json());
app.use(express.urlencoded({ extended: false }));
app.use(cookieParser());

app.use('/', indexRouter);
app.use('/', registrationRouter);
app.use('/', authorizationRouter);

// catch 404 and forward to error handler
app.use(function(req, res, next) {
  next(createError(404));
});

// error handler
app.use(function(err, req, res, next) {

```

```
// set locals, only providing error in development
res.locals.message = err.message;
res.locals.error = req.app.get('env') === 'development' ? err : {};
```

```
// render the error page
res.status(err.status || 500);
res.render('error');
});
```

```
//module.exports = app;
```

```
app.listen(3000);
```

layout.hbs

```
<!doctype html>
<html lang="en">
  <head>
    <!-- Required meta tags -->
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

    <!-- Bootstrap CSS -->
    <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css"
integrity="sha384-
Vkoo8x4CGsO3+Hhxv8T/Q5PaXtkKtu6ug5TOeNV6gBiFeWPGFN9MuhOf23Q9Ifjh"
crossorigin="anonymous">

    <link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.8.2/css/all.css">
    <link rel="stylesheet" type="text/css" href="/stylesheets/style.css" />
    <title>Анализ текста</title>
  </head>
  <body>
    <body>
      {{> menu}}

      {{{body}}}

      {{> footer}}
    </body>
    <!-- Optional JavaScript -->
    <!-- jQuery first, then Popper.js, then Bootstrap JS -->
    <script src="https://code.jquery.com/jquery-3.5.1.min.js"></script>
    <script src="/javascripts/allwords.js"></script>
  </html>
```

footer.hbs

```
<footer class="footer">
```

```
<div class="container">
  <span class="text-muted">Наши контакты: owerchenko.violetta@yandex.ru</span>
</div>
</footer>
```

menu.hbs

```
<header>
  <nav class="navbar navbar-expand-lg navbar-light bg-light">
    <a class="navbar-brand" href="/">Анализ учебных материалов</a>
    <button class="navbar-toggler" type="button" data-toggle="collapse" data-
target="#navbarSupportedContent" aria-controls="navbarSupportedContent" data-
expanded="false" aria-label="Toggle navigation">
      <span class="navbar-toggler-icon"></span>
    </button>

    <div class="collapse navbar-collapse" id="navbarSupportedContent">
      <ul class="navbar-nav mr-auto">
        <li class="nav-item">
          <a class="nav-link" href="/analyses2document">Расчёт совместимости</a>
        </li>
      </ul>

      <div class="text-right">
        <a class="btn btn-light" href="/registration" role="button">Регистрация</a>
        <a class="btn btn-light" href="/authorization" role="button">Вход</a>
      </div>
    </div>
  </nav>
</header>
```

index.hbs

```
<!-- Разработка web-интерфейса для проектирования учебных материалов с учетом их
внутренней и внешней совместимости -->
```

```
<div class="top-message_pages text-center">
  <p id="myhead">Проверка учебных материалов на их внутреннюю и внешнюю
совместимость online</p>
</div>

<div class="container-fluid">

  <div class="row">
    <div class="col-sm-6">
      <div class="card">
        <div class="card-body">
          <form>
            <div class="form-group">
```

```

    <h5 class="card-title">Выберите файл для выделения ключевых слов (pdf, doc,
odt):</h5>
    <!--<label for="exampleFormControlFile1">Выберите файл для выделения
ключевых слов (pdf, doc, odt):</label>-->
    <input type="file" class="form-control-file" id="exampleFormControlFile1">
</div>
</form>

<div class="mb-3">
    <textarea class="form-control" id="exampleFormControlTextarea1"
placeholder="Вставьте текст для анализа" rows="15" required></textarea>
    <div class="text-right">
        <button class="btn btn-primary" type="submit"
id="analyses">Анализировать</button>
    </div>
</div>
</div>
</div>
</div>
<div class="col-sm-6">
<div class="card">
<div class="card-body">
    <h5 class="card-title">Выберите ключевые слова:</h5>
    <div class="table-wrapper-scroll-y my-custom-scrollbar">
        <table class="table table-bordered table-striped table-hover">
            <thead>
                <th scope="col">Ключевое слово</th>
                <th></th>
                <th></th>
            </thead>
            <tbody>

                {{ #each words }}
                <tr>
                    <td>{{ this.descriptionKeyword }}</td>
                    <td width="15px">
                        <a href="/edit/{{ this.idKeyword }}"> <button class="far fa-edit btn btn-
warning"></button></a>
                    </td>
                    <td width="15px">
                        <a href="/deleteWord/{{ this.idKeyword }}"> <button class="far fa-trash-alt btn
btn-danger"></button></a>
                    </td>
                </tr>
                {{ /each }}

            </tbody>
        </table>
    </div>
    <div class="card-body">
        <div class="table-wrapper-scroll-y my-custom-scrollbar">
            <table class="table table-bordered table-striped table-hover">
                <thead>
                    <th scope="col">Ключевое слово</th>
                    <th></th>
                    <th></th>
                </thead>
                <tbody>

                    {{ #each words }}
                    <tr>
                        <td>{{ this.descriptionKeyword }}</td>
                        <td width="15px">
                            <a href="/edit/{{ this.idKeyword }}"> <button class="far fa-edit btn btn-
warning"></button></a>
                        </td>
                        <td width="15px">
                            <a href="/deleteWord/{{ this.idKeyword }}"> <button class="far fa-trash-alt btn
btn-danger"></button></a>
                        </td>
                    </tr>
                    {{ /each }}

                </tbody>
            </table>
        </div>
    </div>
</div>
</div>

```



```

<h1 class="h3 mb-3 font-weight-normal">Регистрация</h1>

<label for="inputEmail" class="sr-only">Введите ваш email</label>
<input type="email" name="email" id="inputEmail" class="form-control"
placeholder="Email" required autofocus>
<label for="inputPassword" class="sr-only">Введите пароль</label>
<input type="password" name="password" id="inputPassword" class="form-control"
placeholder="Пароль" required>
<label for="inputPasswordRepeat" class="sr-only">Повторите пароль</label>
<input type="password" name="passwordConfirm" id="inputPasswordRepeat" class="form-
control" placeholder="Повторите пароль" required>
<span id="message"></span>
<button class="btn btn-lg btn-primary btn-block"
type="submit">Зарегистрироваться</button>
<div class="container signin">
<p>Уже есть аккаунт? <a href="/authorization" >Войти</a>.</p>
</div>
</form>
</div>

```

authorization.hbs

```

<link rel="stylesheet" type="text/css" href="/stylesheets/style_authorization.css" />
<div class="text-center">
<div class="container">
<form class="form-signin" method="POST" action="/authorization" >

<h1 class="h3 mb-3 font-weight-normal">Вход</h1>
<label for="inputEmail" class="sr-only">Email</label>
<input type="email" name="email" id="inputEmail" class="form-control"
placeholder="Email" required autofocus>
<label for="inputPassword" class="sr-only">Пароль</label>
<input type="password" name="password" id="inputPassword" class="form-control"
placeholder="Пароль" required>
<div class="checkbox mb-3">
<label>
<input type="checkbox" value="remember-me"> Запомнить
</label>
</div>
<button class="btn btn-lg btn-primary btn-block" type="submit">Войти</button>
</form>
</div>
</div>

```

edit.hbs

```

<link rel="stylesheet" type="text/css" href="/stylesheets/style_edit.css" />

<div class="container">

{{#each wordData}}

```

```

<h4>Редактирование ключевого слова </h4>
<form class="form-horizontal" action="/updateWord/{{ this.idKeyword }}" method="post">
  <input type="word" class="form-control" id="exampleInputWord" placeholder="Введите
ключевое слово" name='descriptionKeyword'
value="{{ this.descriptionKeyword }}">
  <div class="text-right">
    <button type="submit" class="btn btn-primary">Сохранить</button>
    <a class="btn btn-primary" href="..." role="button">Отмена</a>
  </div>
</form>
{{ /each }}
</div>

```

Analyses2document.hbs

```

<link rel="stylesheet" type="text/css" href="/stylesheets/style_analyses2document.css" />
<div class="container-fluid">

```

```

  <div class="row">
    <div class="col-sm-6">
      <div class="card">
        <div class="card-body text-center">
          <p><h5>Выберите учебный материал</h5><select name="list1">
            <option>Выберите из списка</option>
            <option>Математика</option>
            <option>География</option>
            <option>Английский язык</option>
            <option>Мат.анализ</option>
          </select></p>
          <p><h5>Выберите тему учебника</h5><select name="list2">
            <option>Выберите тему</option>
            <option>Интегрирование</option>
            <option>Дифференцирование</option>
          </select></p>
          <h5 class="card-title text-left">Ключевые слова:</h5>
          <div class="table-responsive">
            <table class="table table-bordered text-left">
              <tbody>
                <tr>
                  <td>Слово1</td>
                </tr>
                <tr>
                  <td>Слово2</td>
                </tr>
                <tr>
                  <td>Слово3</td>
                </tr>
              </tbody>
            </table>
          </div>

```

```

        </table>
    </div>
    <a      class="btn      btn-primary"      href="../pages/analyses2document.html"
role="button">Скрыть ключевые слова</a>
    </div>
</div>
</div>
<div class="col-sm-6">
    <div class="card">
        <div class="card-body text-center">
            <p><h5>Выберите учебный материал</h5><select name="list1">
                <option>Выберите из списка</option>
                <option>Математика</option>
                <option>География</option>
                <option>Английский язык</option>
                <option>Мат.анализ</option>
            </select></p>
            <p><h5>Выберите тему учебника</h5><select name="list2">
                <option>Выберите тему</option>
                <option>Интегрирование</option>
                <option>Дифференцирование</option>
            </select></p>
            <a      class="btn      btn-primary"      href="../pages/analyses2document.html"
role="button">Показать ключевые слова</a>
        </div>
    </div>
</div>
</div>

```

```

<div class="row">
    <div class="col-sm">
        <div class="card">
            <button      type="button"      class="btn      btn-primary"      data-toggle="modal"      data-
target="#exampleModalCenter">
                Рассчитать совместимость
            </button>

            <!-- Modal -->
            <div class="modal fade" id="exampleModalCenter" tabindex="-1" role="dialog" aria-
labelledby="exampleModalCenterTitle" aria-hidden="true">
                <div class="modal-dialog modal-dialog-centered" role="document">
                    <div class="modal-content">
                        <div class="modal-header">
                            <h5 class="modal-title" id="exampleModalCenterTitle">Результат анализа</h5>
                            <button type="button" class="close" data-dismiss="modal" aria-label="Close">
                                <span aria-hidden="true">&times;</span>

```

```

        </button>
    </div>
    <div class="modal-body">
        Совместимость учебных материалов 85%.
    </div>
    <div class="modal-footer">
        <button type="button" class="btn btn-primary" data-
dismiss="modal">Закрыть</button>
        <button type="button" class="btn btn-primary">Готово</button>
    </div>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
</div>

```

package.json

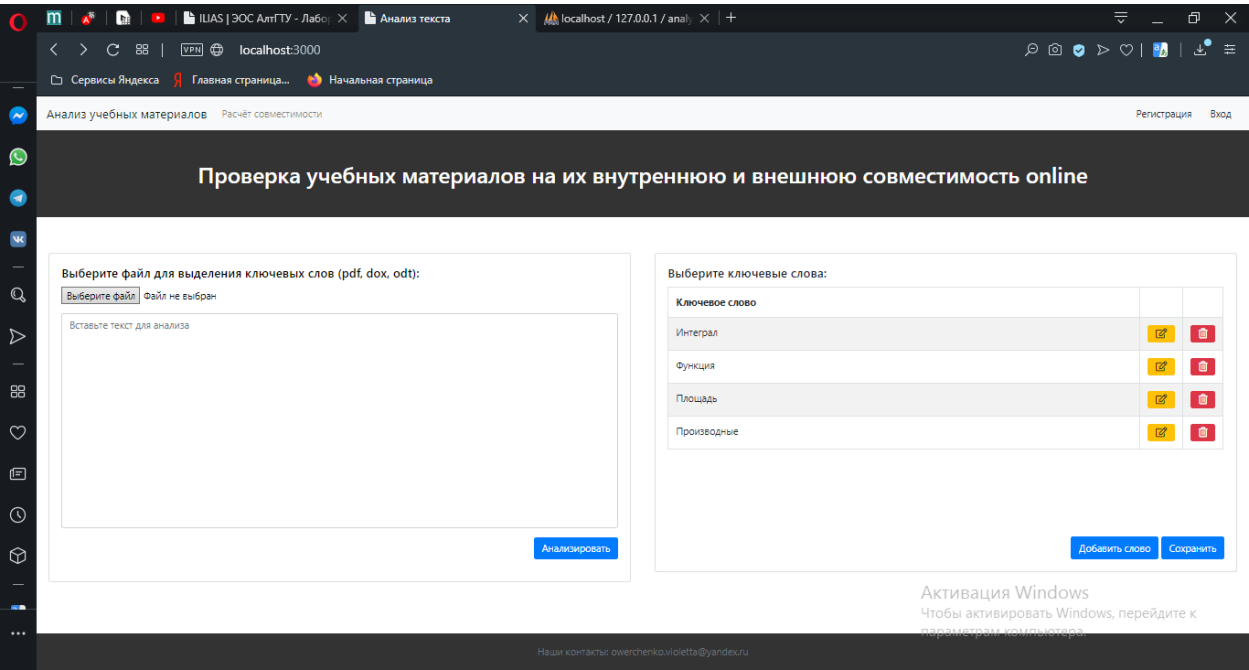
```

{
  "name": "node-js2",
  "version": "0.0.0",
  "private": true,
  "scripts": {
    "start": "node ./bin/www"
  },
  "dependencies": {
    "bcrypt": "^4.0.1",
    "body-parser": "^1.19.0",
    "cookie-parser": "~1.4.4",
    "crypt": "^6.0.2",
    "debug": "~2.6.9",
    "express": "~4.16.1",
    "express-handlebars": "^4.0.3",
    "express-session": "^1.17.1",
    "http-errors": "~1.6.3",
    "morgan": "~1.9.1",
    "mysql": "^2.18.1",
    "nodemon": "^2.0.3",
    "sequelize": "^5.21.7"
  }
}

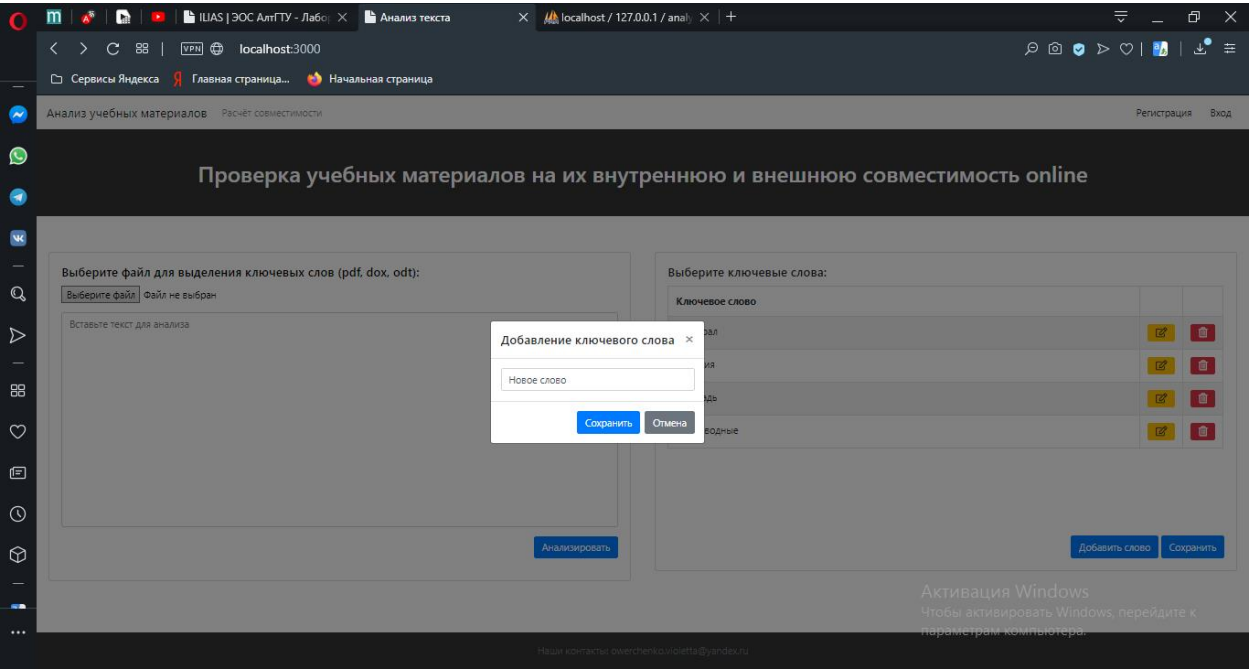
```

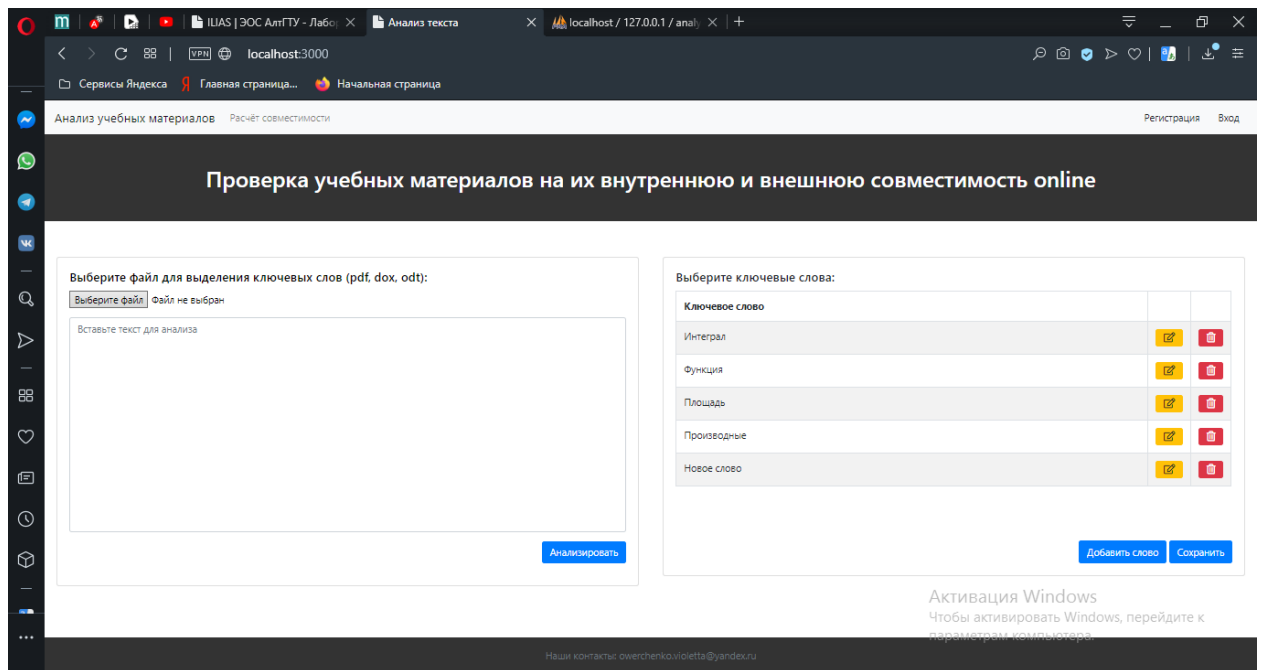
Приложение Б. Снимки пользовательского интерфейса

Read:

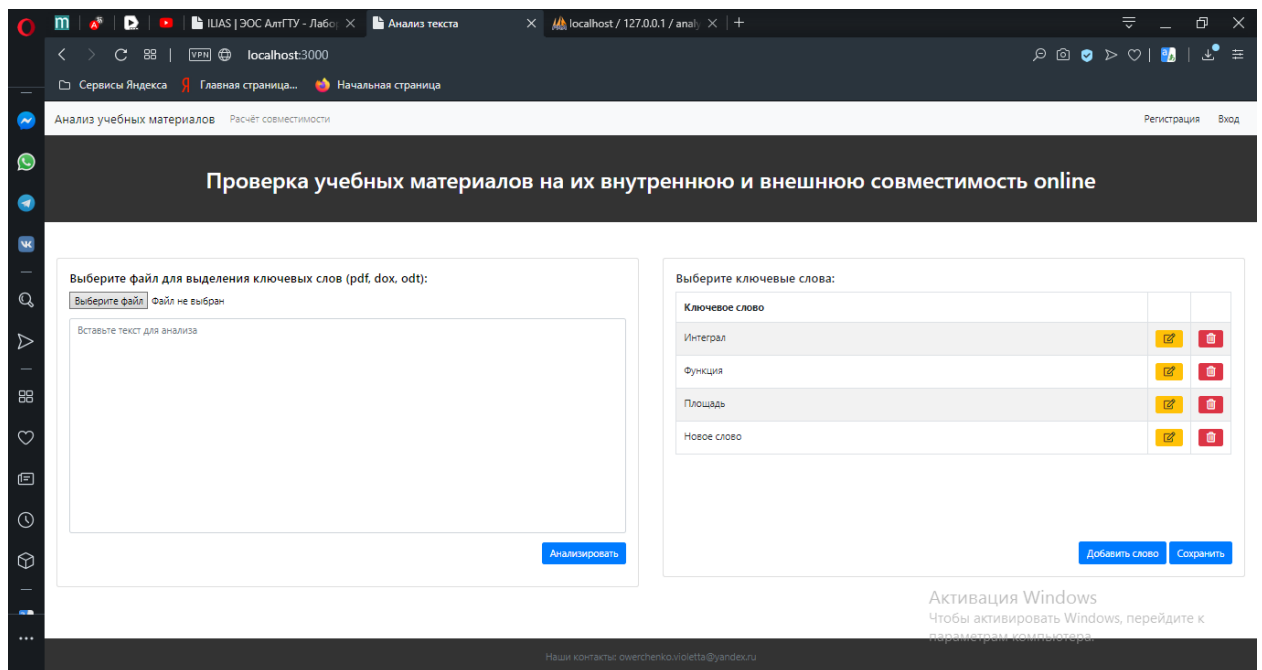


Create:





Delete (Производные):



Update:

