МАТЕМАТИЧЕСКА ГИМНАЗИЯ “АКАД. КИРИЛ ПОПОВ” - ПЛОВДИВ

4002 Пловдив, ул.”Чемшир” № 11, тел.:032/643 157; e-mail: omg@omg-bg.com

**НЕИТ "Джон Атанасов"**

**ИМЕ:**

**EuroDict**

НАПРАВЛЕНИЕ: Разпределени Приложения

**АВТОР:**

***Любомир Владимиров Романов***

Телефон: 0882852973

Е-mail: lr36110213@edu.mon.bg

Училище: МГ „Акад. Кирил Попов”

Клас: 11В

**РЪКОВОДИТЕЛ:**

***Гергина Тодорова Гешева***

Телефон: 0889898399

Е-mail: gergina.gesheva@edu.mon.bg

Длъжност: Учител по информатика и ИТ

СЪДЪРЖАНИЕ:

1. Цели
2. Основни етапи в реализирането на проекта
3. Ниво на сложност на проекта
4. Логическо и функционално описание на решението
5. Реализация
6. Описание на приложението
7. Заключение

**1. Цели**

Проектът **"EuroDict"** представлява самообучаваща програма за учене на италиански, испански и немски език със свободен достъп в интернет. Създаден е с цел да представи по един интересен и атрактивен начин изучавания материал в часовете в училище.

Много от учениците прекарват голяма част от свободното си време пред компютъра и с помощта на софтуерният продукт могат да се съчетаят полезното с приятното.

Graphical user interface

Description automatically generated

**2. Основни етапи в реализирането на проекта**

При реализирането на проекта преминах през следните етапи:

* Избор на тема:

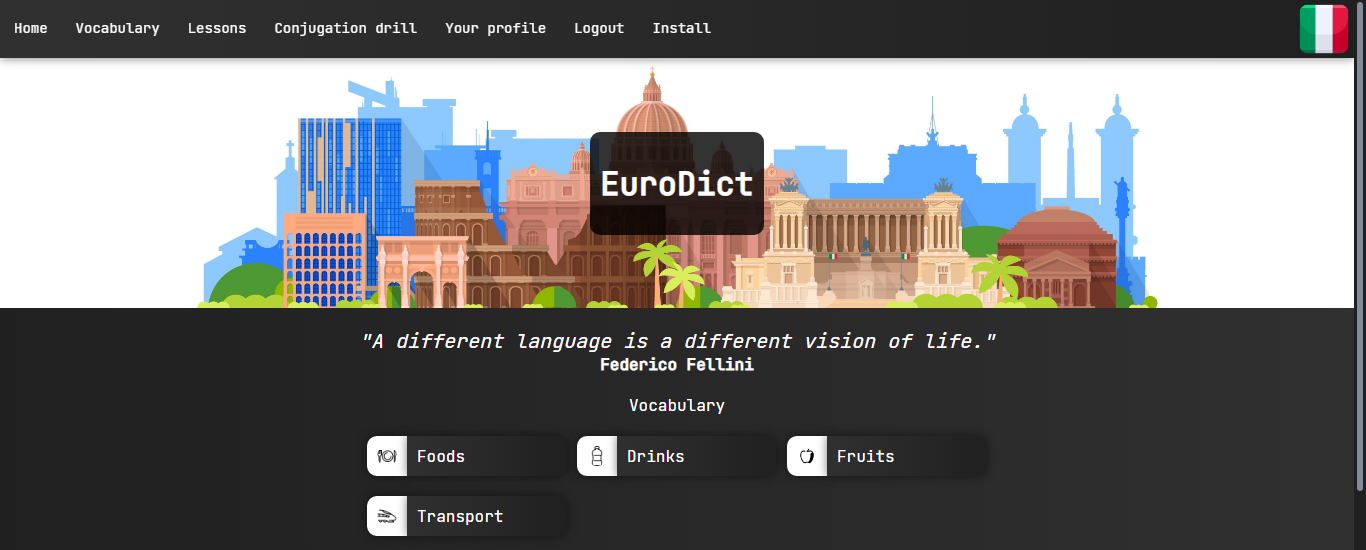
Спрях се на тази тема, защото за мен беше много интересно да науча повече за чуждите езици. Започнах приложението с италиански, постепенно добавих другите два.

* Подбор на материали:

Потърсих допълнителна информация по темата в интернет. Събрах необходимите ми данни и графични изображения.

Обобщих събраните данни и подбрах най-подходящите от тях.

* Определяне структурата на приложението:



* Изработване:

а) Определяне на дизайн – избор на цвят за фон и цвят на символите.

б) Създаване на отделните страници – подредих изображенията и текста.

в) Задаване на хипервръзки между страниците.

* Тестване: Проверих как работят хипервръзките, базата от данни и др. Отворих приложението с различни браузъри, за да проверя дали то е съвместимо.
* Трябваше ми да избера framework. Избрах Flask, защото Python вече беше езикът, на който съм писал най-много код.
* Отстраняване на грешки: Редактирах тези части от страниците, в които имаше неточности.

**3. Ниво на сложност на проекта** – част от трудностите, които срещнах са:

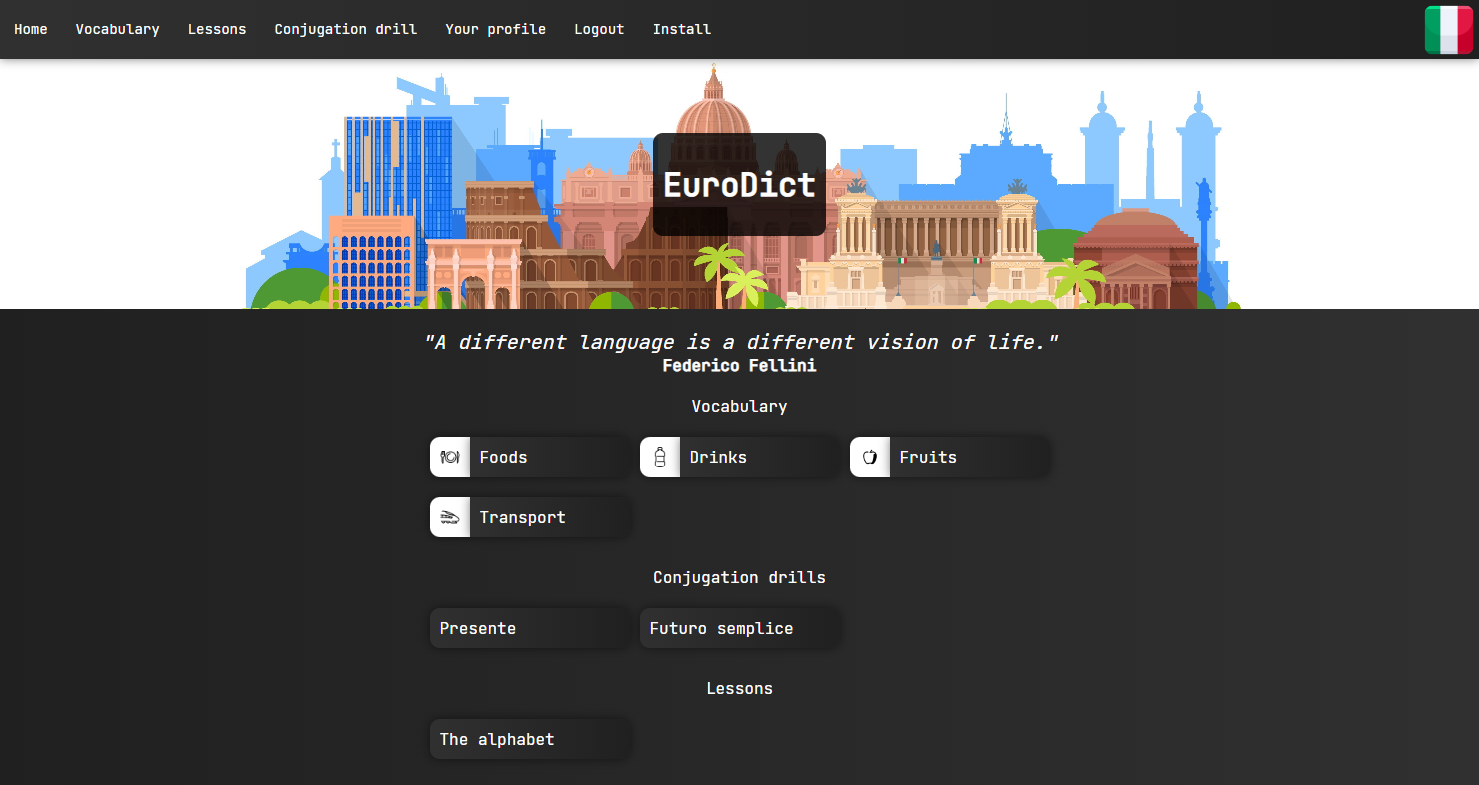
- Огранизиране и създаване на базата данни (SQLite и PostgreSQL)

- Качване на базата данни в интернет

**4. Логическо и функционално описание на решението**:

Проектът представлява самоучител чужди езици представен в интернет. Той е създаден за големите еропейски и световни езици, говорени в целя свят.

В горната част на първата страница е разположено меню, чрез което се осъществява достъп до отделните страници на приложението.



Приложение разполага с лого и име, които се показват във всяка страница в tab-а на браузъра.

**Регистрация на потребител:**

A screenshot of a computer

Description automatically generated with medium confidenceПри регистрация се задава име, имейл, парола и избран език, който може да се смени по всяко време.

1. **Реализация** - За създаването на приложението използвах езиците: HTML, CSS, JavaScript, Python и SQL.

За IDE използвах Visual Studio Code.

1. **Описание на приложението:**

Приложението се стартира от: <http://eurodict.herokuapp.com/>

**Източници на информация** - За изработване на приложението съм използвал следните източници на информация:

<https://www.verbformen.de/>

<https://www.spanishdict.com/>

Ето част от кода

@app.route('/')

def index():

    if current\_user.is\_authenticated:

        if current\_user.language == "italian":

            return render\_template(

                'index.html',

                current\_user=current\_user,

                topics=topics,

                tenses=italian\_tenses,

                tenses\_formatted=italian\_tenses\_formatted,

                lessons\_list=italian\_lessons,

            )

        elif current\_user.language == "german":

            return render\_template(

                'index.html',

                current\_user=current\_user,

                topics=topics,

                tenses\_formatted=german\_tenses\_formatted,

                tenses=german\_tenses,

                lessons\_list=german\_lessons,

            )

        elif current\_user.language == "spanish":

            return render\_template(

                'index.html',

                current\_user=current\_user,

                topics=topics,

                tenses\_formatted=spanish\_tenses\_formatted,

                tenses=spanish\_tenses,

                lessons\_list=spanish\_lessons,

            )

    else:

        return render\_template(

            'index.html',

            current\_user=current\_user,

        )

@app.route('/vocabulary\_topics')

@logged\_in

def vocabulary\_topics():

    return render\_template("vocabulary\_topics.html", current\_user=current\_user)

@app.route('/lessons')

@logged\_in

def lessons():

    if current\_user.language == "italian":

        return render\_template(

            "lessons.html",

            lessons=italian\_lessons,

            current\_user=current\_user

        )

    if current\_user.language == "spanish":

        return render\_template(

            "lessons.html",

            lessons=spanish\_lessons,

            current\_user=current\_user

        )

    if current\_user.language == "german":

        return render\_template(

            "lessons.html",

            lessons=german\_lessons,

            current\_user=current\_user

        )

@app.route('/lesson\_for\_topic/<topic>')

@logged\_in

def lesson\_for\_topic(topic):

    return render\_template(f"lessons/{current\_user.language}/{topic}.html", current\_user=current\_user)

@app.route('/conjugation\_drill\_tenses')

@logged\_in

def conjugation\_drill\_tenses():

    if current\_user.language == "italian":

        return render\_template(

            "conjugation\_drill\_tenses.html",

            tenses\_formatted=italian\_tenses\_formatted,

            tenses=italian\_tenses,

            current\_user=current\_user

        )

    if current\_user.language == "german":

        return render\_template(

            "conjugation\_drill\_tenses.html",

            tenses\_formatted=german\_tenses\_formatted,

            tenses=german\_tenses,

            current\_user=current\_user

        )

    if current\_user.language == "spanish":

        return render\_template(

            "conjugation\_drill\_tenses.html",

            tenses\_formatted=spanish\_tenses\_formatted,

            tenses=spanish\_tenses,

            current\_user=current\_user

        )

@app.route('/conjugation\_drill\_for\_tense/<tense>')

@logged\_in

def conjugation\_drill\_for\_tense(tense):

    if current\_user.language == "italian":

        return render\_template(

            "conjugation\_drill\_for\_tense.html",

            verbs=verbs\_italian,

            tense=tense,

            current\_user=current\_user

        )

    elif current\_user.language == "german":

        return render\_template(

            "conjugation\_drill\_for\_tense.html",

            verbs=verbs\_german,

            tense=tense,

            current\_user=current\_user

        )

    elif current\_user.language == "spanish":

        return render\_template(

            "conjugation\_drill\_for\_tense.html",

            verbs=verbs\_spanish,

            tense=tense,

            current\_user=current\_user

        )

1. **Заключение**

Софтуерното приложение може да бъде полезно, както за обучение в часовете в училище, така и за самообучение.

Чрез него би могъл да се проведе един по-различен учебен час от традиционните, което ще засили интереса на учениците и би довело до по-добри резултати.

Самоучителят беше използван от учителите в Математическа гимназия "Акад. Кирил Попов" в часовете по Немски език 9 и 10 клас.

За в бъдеше софтуерното приложение ще се подобрява като се добавят нови материали.