

Міністерство освіти і науки України  
Департамент освіти і науки Закарпатської  
облдержадміністрації  
Закарпатське територіальне відділення МАН  
України

# СТВОРЕННЯ ВЛАСНОЇ RESTful API ПЛАТФОРМИ ТА ІНТЕРАКТИВНОГО ЩОДЕННИКА НА ЇЇ ОСНОВІ

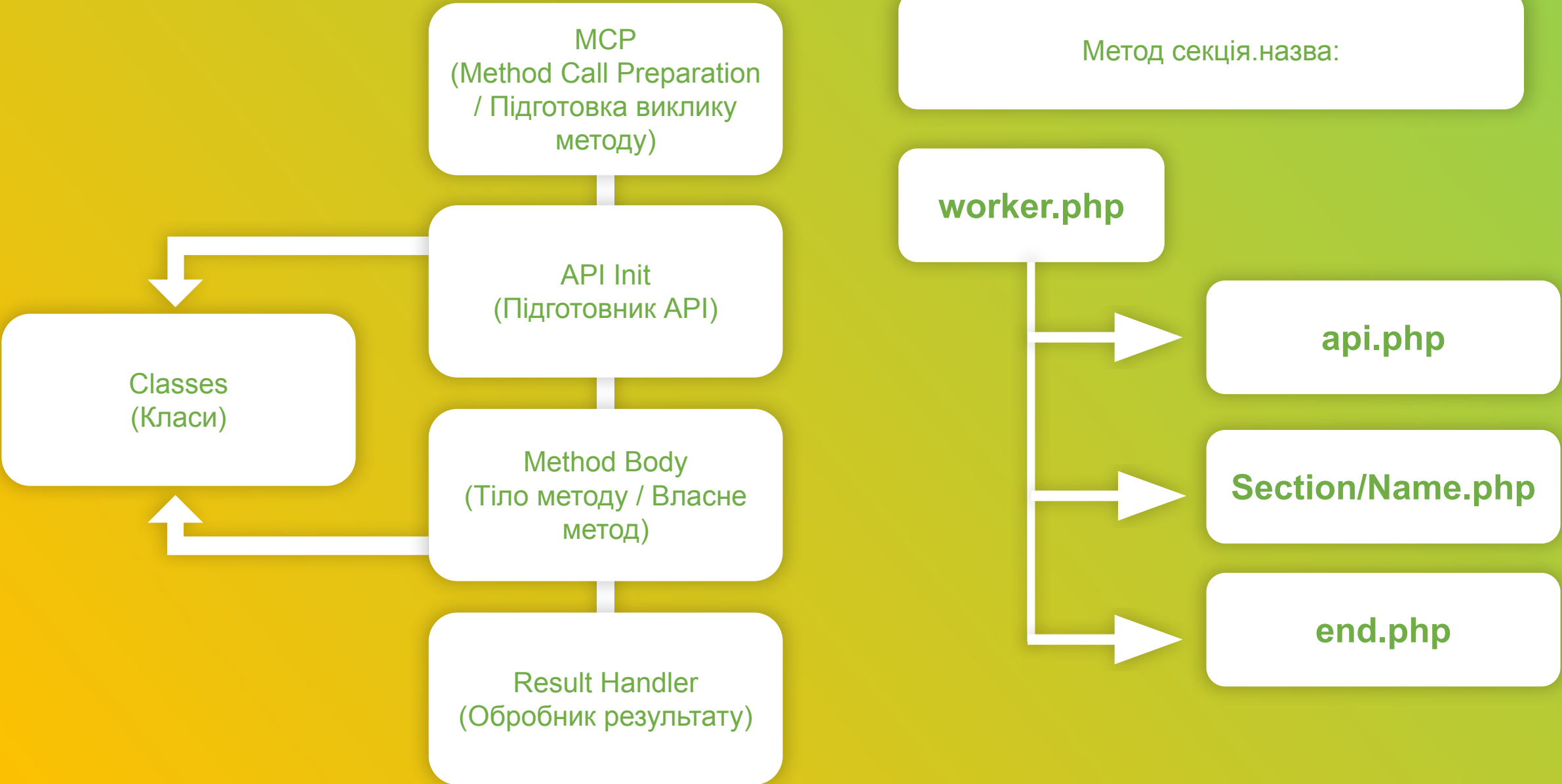
Відділення: Комп'ютерних наук  
Секція: Internet- технології та WEB дизайн  
Роботу виконав:  
Ділонг Сергій Михайлович  
учень 11 класу  
Ужгородської класичної гімназії



- **Метою роботи** є проведення пошуково-дослідницької роботи з питань пов'язаних із створенням RESTful API платформи, вивчення потреб учнів для подолання невідповідності пропонованих продуктів для контролю часу.
- **Отримані результати:** систематизовано, узагальнено та проведено пошуково-дослідницьку роботу щодо створення універсальної платформи для сервісів, зроблено прототип, платформу документації.

- API – спосіб взаємодії двох абсолютно різних або схожих незалежних структур (у моєму випадку – сервіс та браузер) для передачі інформації та виконання дій.
- RESTful API – філософія створення API, за якою використовуються усі методи та коди статусу HTTP для передачі інформації

# Схема роботи



account.shareByCode.php

contacts.find.php

contacts.get.php

messages.get.php

test.server.php



methods

Account

Contacts

Dev

Messages

Get.php

Test

# Документація

Секція

Помилка

Метод

Клас

Глобальн  
ий  
параметр



### Methods

Account

Contacts

Messages

Test

Token

Users

### API

Classes

Global variables



## Classes

Objects having exact properties everywhere in API

Chat class – Describes Chat object

Message class – Describes Message object

Pages object – Versatile pagination mechanism

User class – Describes profile object



## users.getByCode

Get users' data by temporary code

Returns array of users that correspondent to temporary code(s), given in [account.shareByCode](#) method.

### Hints



The object could require authentication

### Request

**codes**

Given users' codes integers, separated by comma, **required field**

### Response

**response**

Result by query array, that contains objects of type [User class](#)

INT

INT  
TIMESTAMP

ARRAY

STRING

BOOL

CLASS

BINARY

COMMA

# GraphQL Markdown

## users.get

*Get users' data by login*

Returns each User data from query field. This method allows to get User ONLY by login. If user doesn't exist `false` will be returned. If query is empty current user will be returned.

## Hints

- [The object could require authentication](#)

## Request

- **query**  
Required users' logins  
*strings, separated by comma*

## Response

- **response**  
Result by query or current user's data  
*array, that contains objects of type [User class](#), **required field***

Ім'я	Опис	Тип
id	Унікальний ID [AUTO_INCREMENT]	INT
name	Назва	VARCHAR (255)
default_cab	Кабінет за вмовчуванням	TEXT

Ім'я	Опис	Тип
id	Унікальний ID [AUTO_INCREMENT]	INT
day	День тижня [1-7]	INT
subject	ID предмета subjects	INT
start_time	Початок заняття у хвилинах від початку доби (0:00)	INT
end_time	Кінець заняття у хвилинах від початку доби (0:00)	INT
cab	Кабінет. Якщо не вказано - використовувати default_cab з таблиці subjects	TEXT

## Add new Element

Day	Tu
Subject	English
Start Time	15:05
End Time	15:50
Cabinet	202
<input type="button" value="Отправить"/>	

- Автозаповнення часу на основі даних при першому налаштуванні
- Автоматичні підказки при заповненні назви предмета
- Автозаповнення кабінету, якщо такий предмет вже існує
- При першому налаштуванні необхідні лише найчастіша довжина перерви, довжина уроків, початок занять

Home

Add Element

Settings

4

07:55 AM

Set Time

Set up your timetable defaults:

Lesson length

45

Break length

10

Lessons start

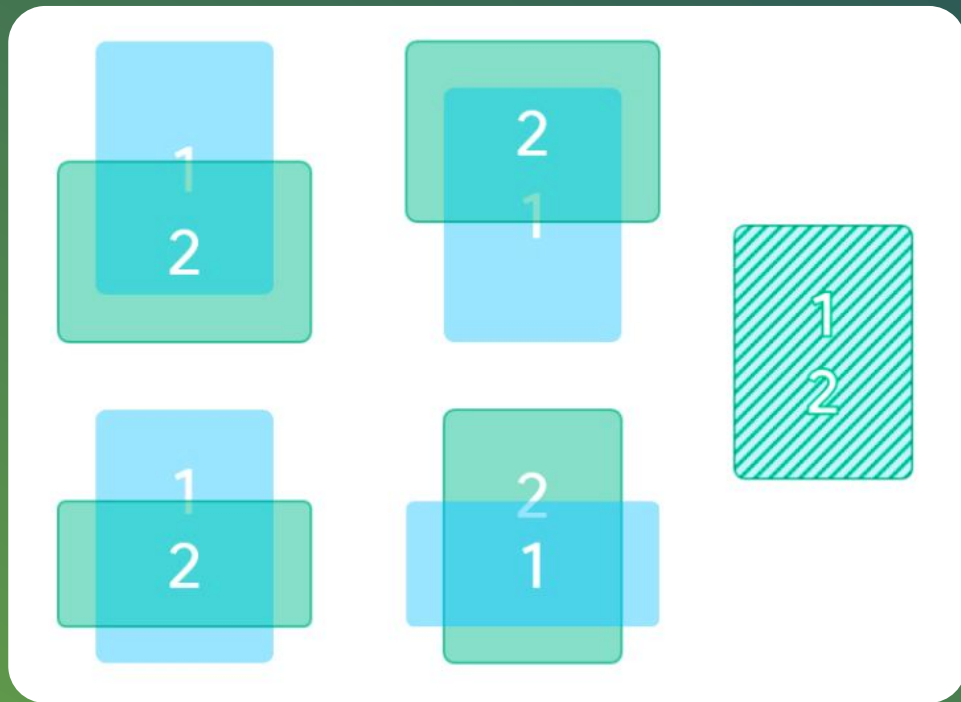
08:30 AM

Submit

Reset

- ▶ I wish to clear all data
- ▶ Restore defaults

# Правильність даних



$S1 < S2$	$S1 > S2$	
$E1 > E2$	$E1 < E2$	
$S1 < E2$	$S1 < E2$	$S1 = S2$
$E1 > S2$	$E1 > S2$	$E1 = E2$
$S1 < S2$	$S1 > S2$	$S1 < E2$
$E1 < E2$	$E1 > E2$	$E1 > S2$
$S1 < E2$	$S1 < E2$	
$E1 > S2$	$E1 > S2$	

15:40

**0:25:37**

Algebra ends at 13:15

**Timetable** TU, 22th March

Algebra	8:30 – 9:15
English	9:25-10:10
History	10:20-11:05
PE	11:25-12:10
Algebra	12:30-13:15

Navigation icons: Profile, Clock, Chat, Home, Back, Forward

15:40

**Today** TU, 22th March

1 Algebra	22
2 English	113
3 History	113
4 PE	GYM
Algebra	22
6 Biology	BC
7 Classroom hour	113

**Tomorrow** WE, 23th March

Navigation icons: Profile, Clock, Chat, Home, Back, Forward

15:40

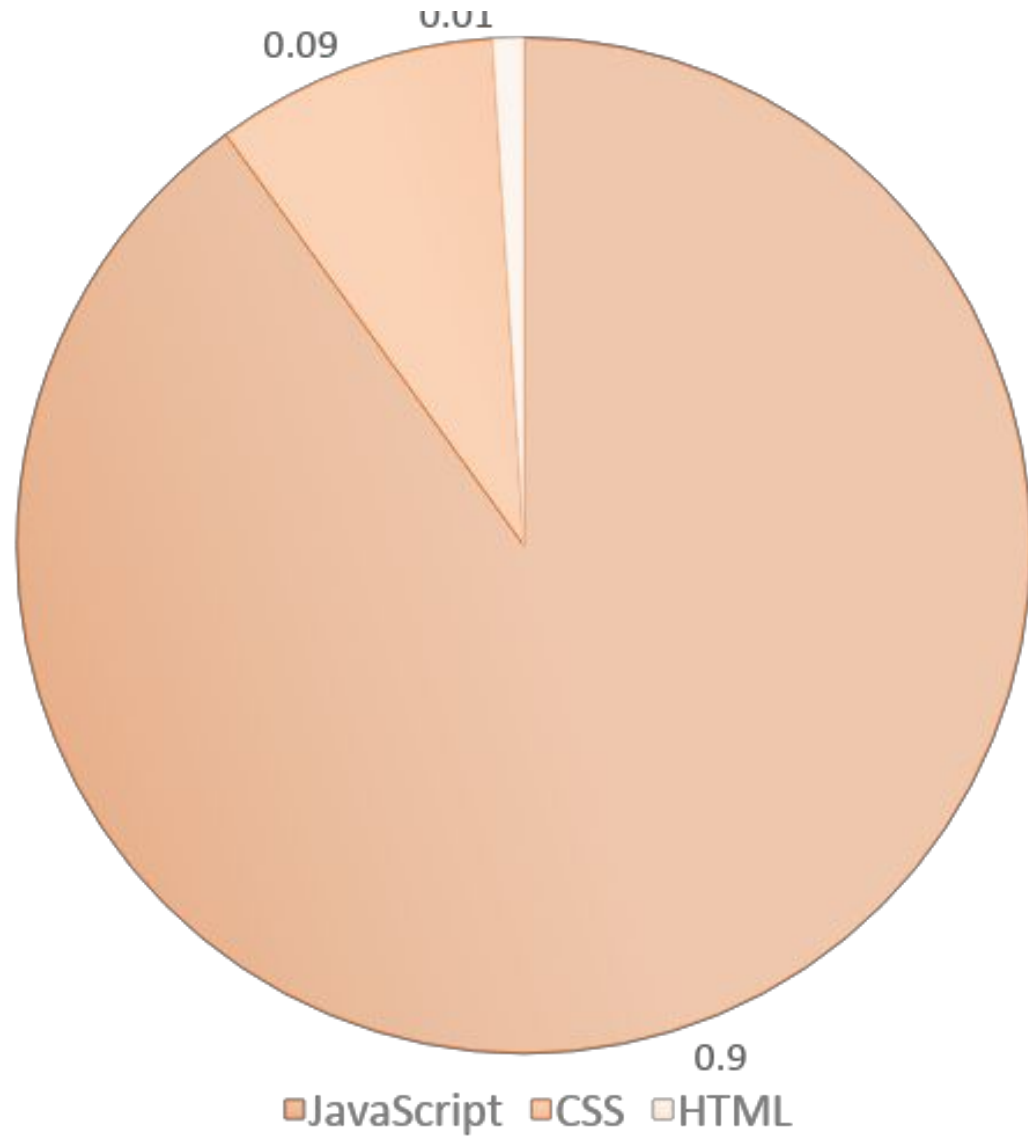
**Today**

- Biology • Kate Smith  
Fill the table in copybook using recommendations in book
- Classroom Hour • You  
Birthday preparation: money and cakes

**Tomorrow**

- Algebra • You  
Presentation: The nth power equation
- IT Technologies • You  
Finish presentation and fill it using Visual Basic for Applications

Navigation icons: Profile, Clock, Chat, Home, Back, Forward



- HTML5
- ES2018
- Webpack
- Babel



Ознаки сайту	Ознаки застосунку
Виконується у браузері	Виконується безпосередньо на рівні ОС
Використання інтерпретованої мови програмування	Використання компільованої мови програмування
Неможливість виконання операцій у фоні	Можливість виконання операцій у фоні
Що разове завантаження ресурсів для відображення інтерфейсу	Одноразове завантаження усіх ресурсів, пов'язаних з інтерфейсом
Немає доступу до надсилання нативних сповіщень користувачу або обмежений доступ до їх функціональності	Можливість відправляти сповіщення, використовуючи усі доступні засоби в ОС
Не потребує встановлення	Потребує встановлення
Може суттєво оновлювати свої компоненти у фоновому режимі	Може суттєво оновлювати свої компоненти лише після встановлення оновлення

# HTML5

Бази даних розмірами у десятки гігабайт

Тісна інтеграція з браузером

# ES2018

Використання спрощеного та ефективного синтаксису

Чітка структуризація коду з принципом модульності та ООП

# Webpack

Оптимізація коду та включення ресурсів для блискавичного завантаження

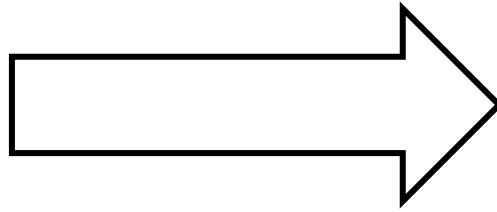
Babel

# Babel

Автоматизован  
а  
кросбраузерність

Використання  
можливостей мови  
програмування до того, як  
їх внесено у специфікацію

- JS** appendContent.js
- JS** attributeSetter.js
- JS** elementConstructor.js
- JS** eventsSetter.js
- JS** genericElement.js
- JS** htmlSet.js
- JS** idSetter.js
- JS** propertySetter.js
- JS** setClasses.js
- JS** valueSetter.js



bundle.js  
[vendor.js]

**Модуль**



**Модифікатор**



**Властивість**



**Подія**



Утворення



Очікування



Рендер

```
export default class Icon {  
  constructor(name) {  
    return new DOM({  
      new: "icon",  
      content: new DOM({  
        type: "t",  
        new: name.toString(),  
      })),  
    })  
  }  
}
```

`new Icon("search")`



Живий приклад

Документація

Застосунок