**Міністерство освіти і науки україни**

**ЗАПОРІЗЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ**

**Математичний факультет**

**Кафедра фундаментальної математики**

ЗВІТ

**З ЛАБОРАТОРНИХ РОБІТ № 2**

З курсу: **«Frond-end технології»**

Виконав: студент 3 курсу, групи 6.1218-2

спеціальності 121 Інженерія програмного забезпечення

Д.О. Переуда

Запоріжжя – 2020

**ЗАВДАННЯ**

1. Використовуючи текстовий редактор5 , створити набір документів: − index.html − rozklad.html − news.html − photo.html

2. Кожна з сторінок повинна мати однакову структура з використанням відповідних семантичних елементів HTML5.

3. Навігація кожної з сторінок повинна мати посилання на всі інші документи перелічені у пункті 1.

4. Макет сторінки index.html повинен відповідати макету наведеному на рис. 2.3 та використовувати як можна більше тегів html для оформлення тексту (цитати, посилання, примітки, переліки і таке інше).

5. Сторінка rozklad.html повинна містити таблицю з розкладом занять поточного семестру та повний маркований перелік викладачів, що викладають дисципліни у Вашій академічній групі. Макет сторінки наведено на рис. 2.3.

6. Сторінка photo.html повинна містити мініатюри 8-10 зображень (усі однакового розміру) з підписами (назва та розмір оригіналу). Кожна з фото повинна бути посиланням на оригінал зображення, який відкривається у окремому вікні браузера. Макет сторінки наведено на рис. 2.4.

7. Сторінка news.html повинна містити анонси (зображення, дата публікації, короткий опис, посилання на повну версію) 5-8 статей-новин (з популярних порталів новин). Посилання повинне відкривати повну версію матеріалу на сторонньому ресурсі. Макет сторінки наведено на рис. 2.4

8. Пов’язані графічні матеріали розташувати поруч з html-документами у окремій директорії – images.

9. Розташувати створені сторінки та зображення на хостингу у окремій директорії (lab2).

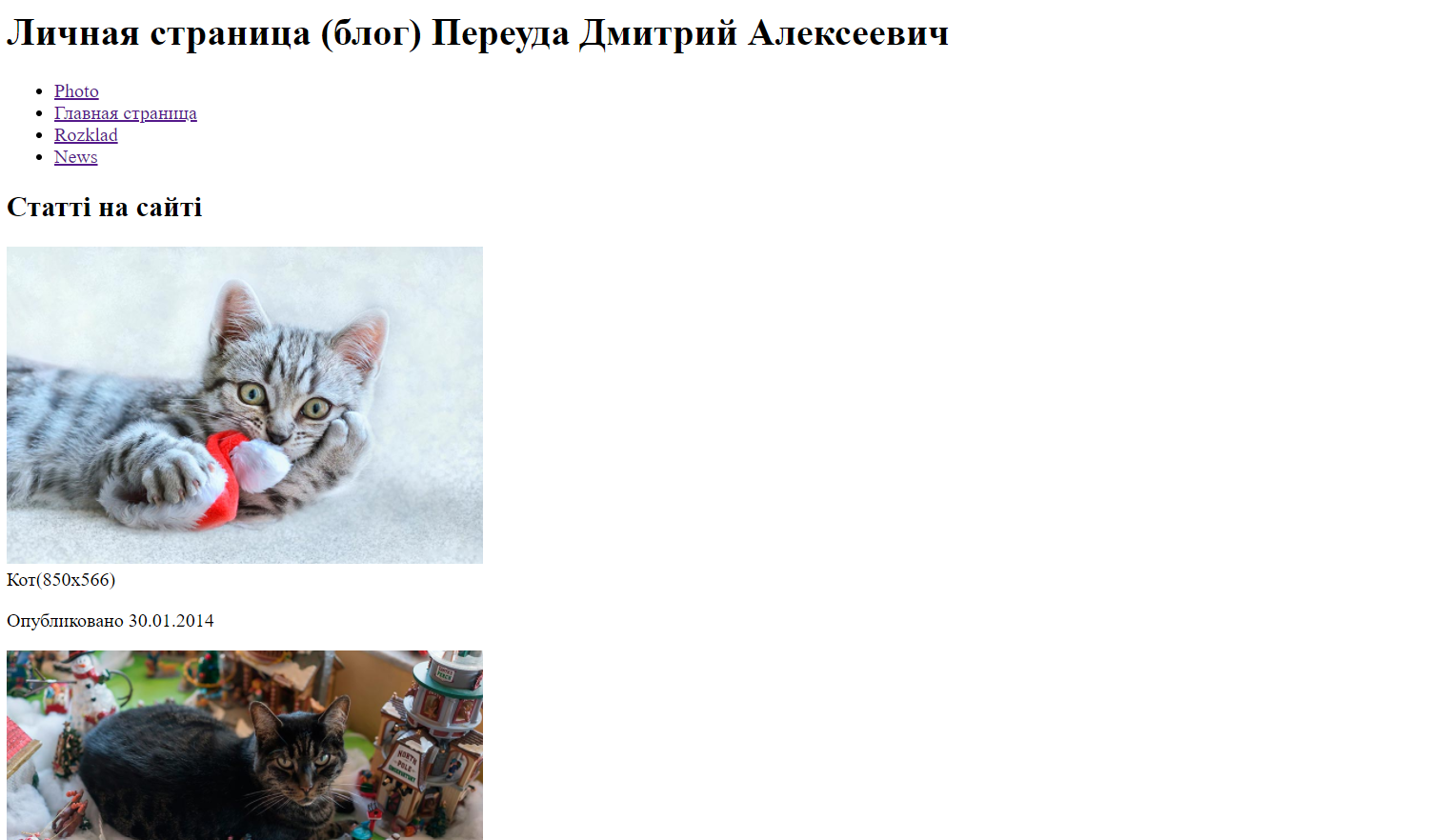
10. Виконати перевірку6 (валідацію) сторінок на відповідність стандартам HTML5.

11. Результати виконання лабораторної роботи розташувати у власному репозиторії будь якої відкритої системи контролю версій з тегом lab2.

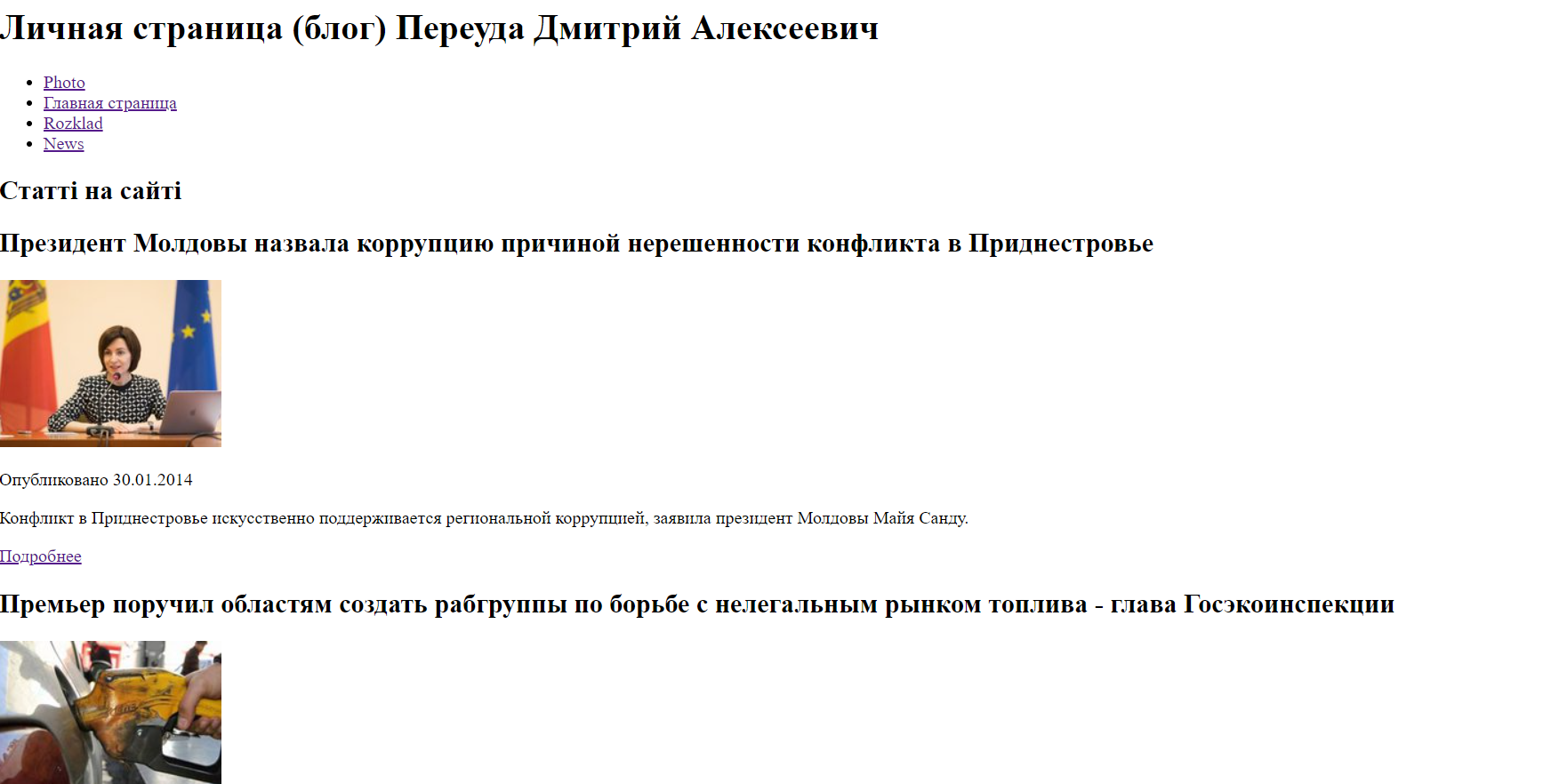
12. Предметами захисту роботи є вихідний текст у репозиторії системи контролю версій та загальнодоступний ресурс, що відображає результат виконання лабораторної роботи.

**ХІД ВИКОНАННЯ РОБОТИ**









**Контрольні запитання**

1. Наведіть приклади тегів, що є застарілими у HTML5.

Различных **тегов устарели** следующие атрибуты: У **тега** <a> **устарели** атрибуты charset=" " , coords=" " , shape=" " , methods=" " , name=" " , rev=" " и urn=" " . Вместо атрибута name=" " , у якорей, нужно использовать атрибут id=" "

1. Які правила використання тегу ?

Согласно спецификации **HTML** все значения атрибутов **тегов** следует указывать в двойных ("пример") или одинарных кавычках ('пример'). Отсутствие кавычек не приведет к ошибкам, браузеры во многих случаях достаточно корректно обрабатывают код и без кавычек, за исключением текста, содержащего пробелы

1. Перелічіть основні теги для визначення семантичної структури документа HTML5.

Елемент **<header>**

Елемент **<hgroup>**

Елемент **<nav>**

Елемент **<article>**

Елемент **<section>**

Елемент **<aside>**

Елемент **<footer>**

Елемент **<address>**

Елемент **<main>**

Елемент **<time>**

Елемент **<mark>**

Елемент **<bdi>**

1. Як браузер визначає версію HTML документа?

HTML и HTTP очень связаны между собой, если вы не знаете, то HTTP протокол синхронный, это означает, что во время [HTTP соединения](https://zametkinapolyah.ru/servera-i-protokoly/tema-8-http-soedinenie-http-connections.html) и клиент, и сервер обмениваются сообщениями по очереди. Сперва клиент отправляет запрос, потом сервер отправляет ответ на запрос, если клиенту нужно, он делает еще запрос, на который опять же отвечает сервер.

HTTP клиент в нашем случае – это браузер. По HTTP протоколу чаще всего передаются HTML документы. Из-за особенностей **реализации HTTP протокола и структуры HTML документа** браузеры реализованы так, что они читают и интерпретируют HTML код слева направо и сверху вниз: сначала браузер проверяет версию HTML, указанную в декларации, затем он видит тэг <html>, который говорит ему о том, что начался HTML документ.

1. Які особливості документів XHTML?

XHTML - это основанный на [XML](https://htmlweb.ru/xml/xml1.php) язык разметки гипертекста, максимально приближенный к текущим стандартам [**HTML**](https://htmlweb.ru/html/bhtml.php). XHTML отличается от HTML строгостью написания кода. Если HTML позволял писать практически любые конструкции и браузер их корректно распознавал, то теперь, с появлением XHTML, это стало невозможным. Последний требует строгого соблюдения всех правил, предъявляемых W3C. Строгие требования к оформлению XHTML-кода позволяют избежать многих ошибок ещё на стадии написания и отладки.

* в теле **XHTML-документа** должны обязательно присутствовать следующие тэги: *html*, *head*, *title* и *body*;
* обязательно наличие атрибута *xmlns* в элементе *html*;
* имена тегов и атрибутов должны записываться в нижнем регистре;
* все значения атрибутов должны заключаться в "кавычки";
* все тэги должны закрываться; если у элемента нет закрывающего тэга, следует добавлять в его конец слеш (*<br/>* или *<br />* — пробел для совместимости со старыми браузерами);
* необходимо соблюдать корректную вложенность тэгов (*<b><i>текст</b></i>* — неверно; следует писать *<b><i>текст</i></b>*);
* запрещается использовать минимизированные атрибуты (*nowrap* следует заменить на *nowrap="nowrap"*); полный список таких атрибутов: *checked*, *compact*, *declare*, *defer*, *disabled*, *ismap*, *noresize*, *noshade*, *nowrap*, *multiple*, *readonly*, *selected*.
* на следующие элементы налагаются ограничения по включению в них других элементов:
  + *a* не может содержать другие элементы *a*;
  + *form* не может содержать другие элементы *form*;
  + *label* не может содержать другие элементы *label*;
  + *pre* не может содержать *img*, *object*, *big*, *small*, *sub* или *sup*;
  + *button* не может содержать элементы *input*, *select*, *textarea*, *label*, *button*, *form*, *fieldset*, *iframe* или *isindex*;
* специальные символы в истинном значении должны заменяться на свои эквиваленты:
  + *&* на *&amp;*;
  + *<* на *&lt;*;
  + *>* на *&gt;*.