**1. Постановка задачи проектирования и стратегическое планирование**

**1.1. Название проекта** — freehands. Тематика данного ресурса: предоставление своего резюме для получения работы и представление новостей из мира WWW.

**1.2. Внутренними целями** данного сайта является представлении собственных навыков и получение работы от работодателей.

**1.3. Внешними целями** является получение информации о IT-сфере и робочих кадров.

**1.4. Целевая аудитория. Целевые запросы.**

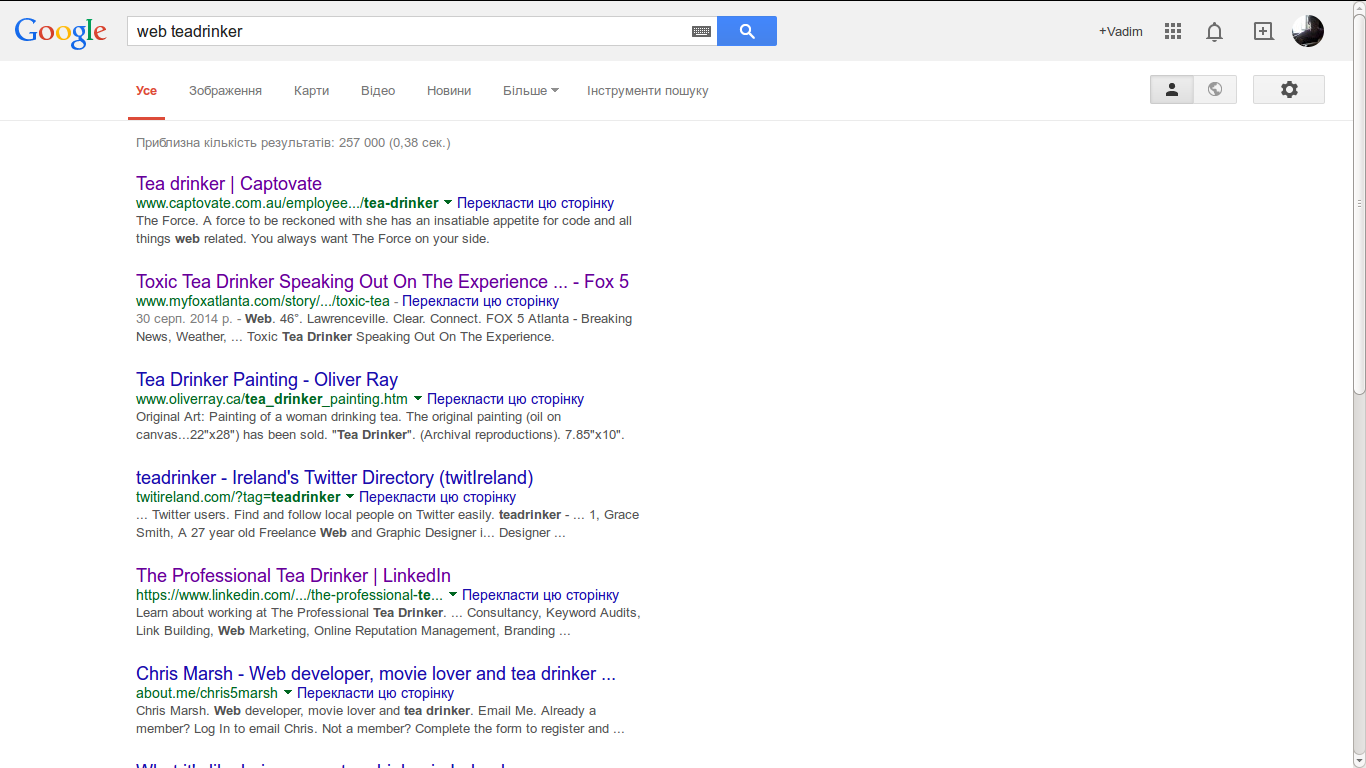
Целевая аудитория росчитана на людей интересующиемся IT-сферой и заказчиков.

Целевые запросы будут связаны с информацией в IT-сфере, а также моими личными успехами.

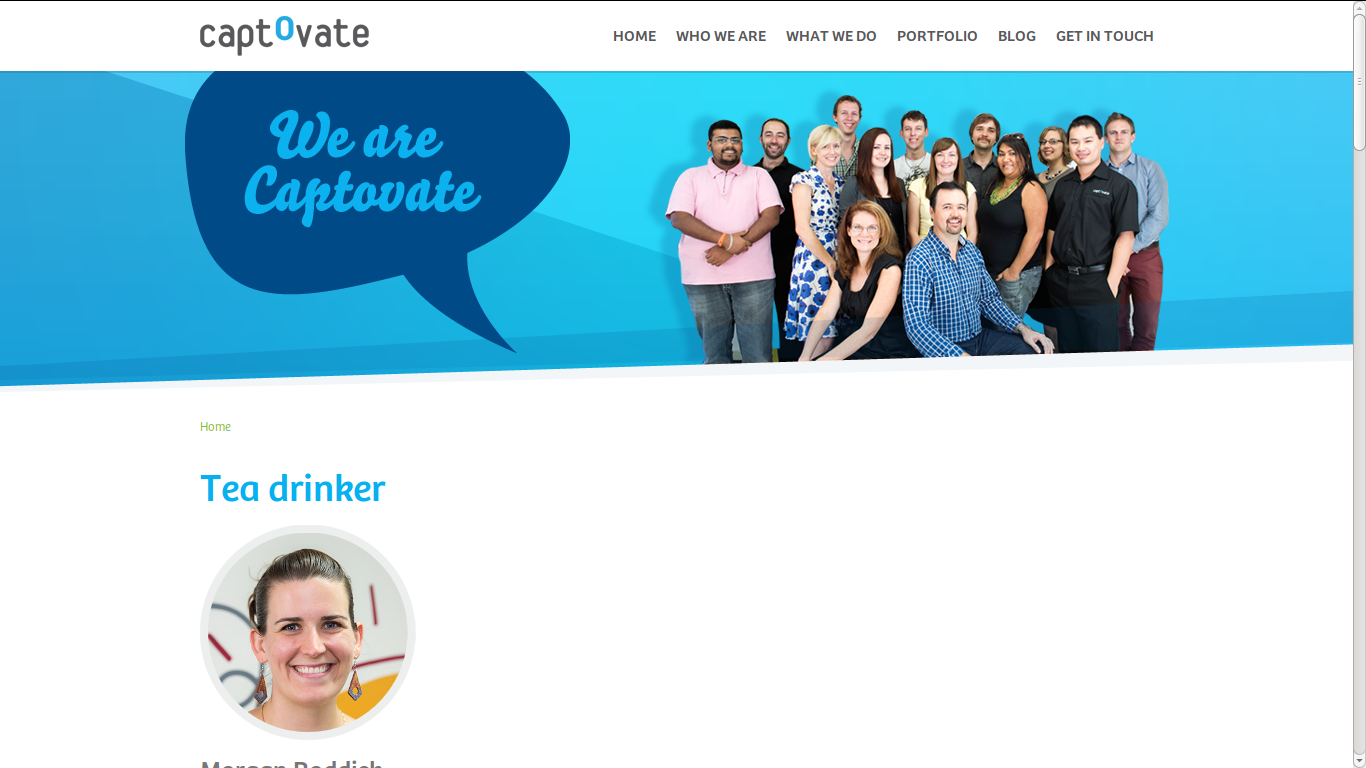
Примеры запросов: «TeaDrinker», «web», «programming», «develop», «dev»….

**1.5. Анализ конкурирующих проектов:**

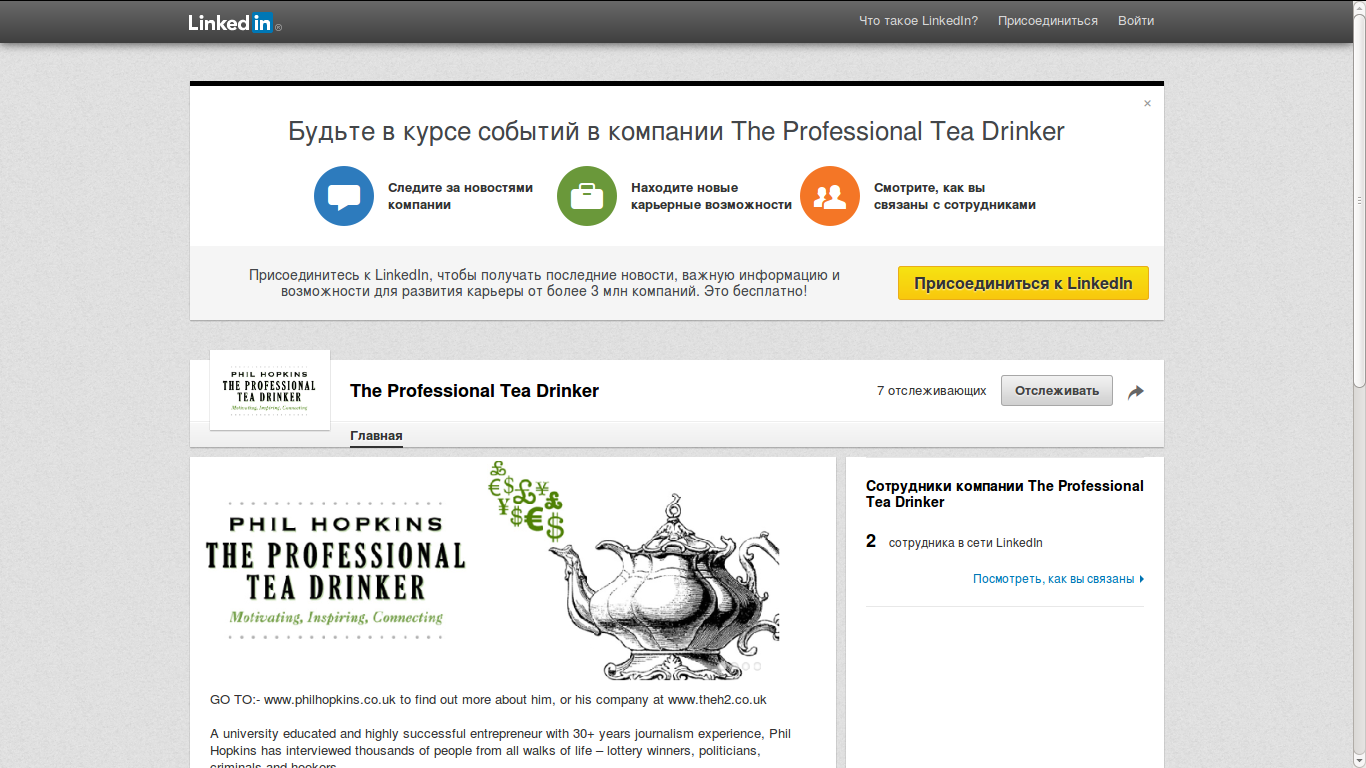
По запросу на «web teadrinker»:



<http://www.captovate.com.au/employee-tags/tea-drinker>

****

<https://www.linkedin.com/company/the-professional-tea-drinker>

<http://www.altima.com.ua/?gclid=Cj0KEQiAwPCjBRDZp9LWno3p7rEBEiQAGj3KJpDKOX6ZE1lzBVK5lHNwhChZQk3BvQ0C3XMn2ao5c-0aAknk8P8HAQ>



|  | *captovate* | *linkedin* | *altima* |
| --- | --- | --- | --- |
| **Дизайн** |  |  |  |
| Удобство восприятия информации | 5 | 4 | 3 |
| Единый стиль шрифтов, цветов, графики | 5 | 5 | 5 |
| Привлекательность | 5 | 5 | 5 |
| **Функции** |  |  |  |
| Наличие поиска | - | - | + |
| Форма обратной связи | + | + | + |
| **Контент** |  |  |  |
| Доступность информации | 5 | 4 | 5 |
| Грамматика | 5 | 5 | 5 |

Существуют очень сильные проекты которые уже имеют свою сферу влияния, но они ориентированны в основном на другой контенгент пользователей.

**1.6. Классификация сайта** — личный проект.

**1.7. Особенности сайта** — одновременное представление информации полезной для пользователя и работодателя, а также выражение собственных навыков.

**2. Стратегическое планирование**

**2.1. Источники финансирования** — личные средства.

**2.2. Обосновать выбор разработчика.** Сайт был спроектирован собственноручно автором проекта.

**2.3. Спецификация контента.** На сайте 4 основных пункта:

1) Main

2) Resume

3) Contacts

3) Blog

Это сайт имеет контактные данные о создателе сайте и его навыки.

**2.4. Обосновать технические требования к серверу, канал связи, статический или динамический сайт, выбор необходимых технологий.**

Платный хостинг, он стоит в районе 3-10$ в месяц, при етом предоставляет тех.поддержку 24/7, PHP, MySQL, Mail, ftp, набор настроек и поддержку разных CMS(WordPress, Joomla).

Одним из таких хостингов является украинский хостинг от [http://www.ukraine.com.ua](http://www.ukraine.com.ua/).

Сайт динамический, реализован на собственном движке.

**3. Разработка информационной архитектуры**

**3.1. Основные разделы:**

Main — в разделе содержится короткая информация обо мне и ссылки на более специфическую информацию.

Resume — в разделе содержанием является общая информация обо мне и мои навыки.

Contact — в разделе содержится форма обратной связи, где гость сайта может напрямую обратиться ко мне с вопросом.

Blog — раздел содержит информацию, выбранной мною, о мире программирования, а так же о моих личных успехах.

**3.2. Схема информационной архитектуры**

В проекте используется гибридная структура.

Main

Resume Contacts Blog Table layout

Skills Web front-end

Works back-end

API

CMS

Administration

Programming

other

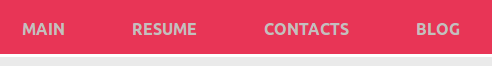
**3.3. Физическая структура.**

Написан мною движок сохраняет статьи в бд, которая находиться на сервере. Информация достается из бд с помощью скрипта и генерируеться на странице. URL при етом читабельн и пользователю проще запомнить категорию и №статьи.

**3.4. Используемые виды навигации, с привязкой к схеме информационной архитектуре.**

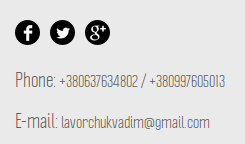
Используется комбинированая структура, так как базовой навигацией являеться плоская структура, а уже в ней используется древовидная структура.

Навигация по сайту:

Внутренне меню:

Ссылки в социальные сети:





**3.5. Учет поисковой оптимизации.**

Навигация разбита по основным разделам. В разделах имеется своя навигация, с логическими названиями и отображением в адресной строке, по етому запомнить адрес не составит больших проблем.

**4. Контент-проектирование**

**4.1. Виды и формы контента.**

В основном, выделяют 2 формы контента:

* Коммерческий контент – это содержание веб-сайтов, занимающихся продажей каких-либо товаров или, услуг.
* Некоммерческим контентом называют развлекательный, новостной, информационный или другой материал, не преследующий финансовых целей. Мой сайт содержит именно некоммерческий контент, так как отсутствует какая-либо продажа, есть только информация.

Кроме этого, контент можно классифицировать по форме. Различают текстовый, звуковой и графический контент. Мы имеем дело именно с текстовым и графическим контентом.

Также разделяют три степени уникальности контента: уникальный, частично уникальный, неуникальный. На сайте частично и уникальный контент, так как статьи пишутся непосредственно разработчиком.

**4.2. Источники контента:**

Текстовый контент, предоставляет основатель сайта, часть информации берется из интернета.

Графический контент, предоставляет основатель сайта, часть информации берется из интернета.

**4.3. Требования поисковой оптимизации.**

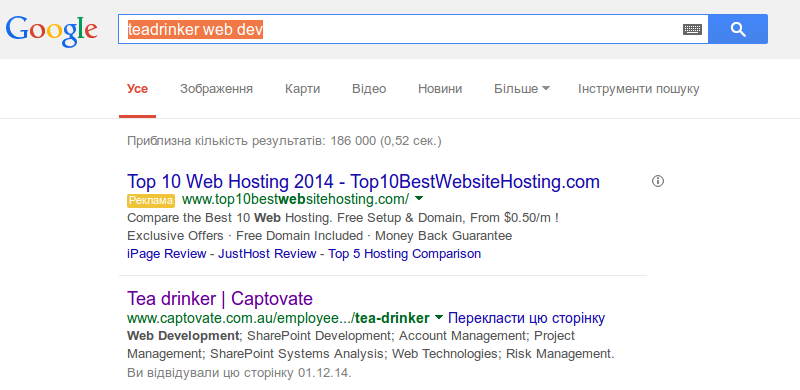
Т.к. наш сайт ничего не продает, то пользователи будут совершать именно информационные запросы.

Наш пользователь — это:

* запрос в поисковой системе;
* переход на одну из страниц сайта;
* путешествие по сайту;
* полная информированность пользователя по его запросу.

Поиск должен осуществляться по ключевым словам, т.е. тем словам(фразам), которые чаще всего употребляются на сайте. Вот пример с такой ключевой фразой.

Пользователь ищет по запросу : «teadrinker web dev»:



Когда пользователь заходит на сайт он видит компанию занимающеюся веб-разработкой, знакомиться с их компанией и извлекает нужную информацию для себя.

**5. Дизайн**

**5.1. Основные типы документов с привязкой к структуре сайта и схеме информационной архитектуре (п.3.).**

Сайт является динамическим так как контент подгружается средствами JavaScript и AJAX.

**5.2. Состав и шаблоны каждого типа документов**

Все страницы сайта имеют один общий шаблон (у таблицы совсем все по-другому).

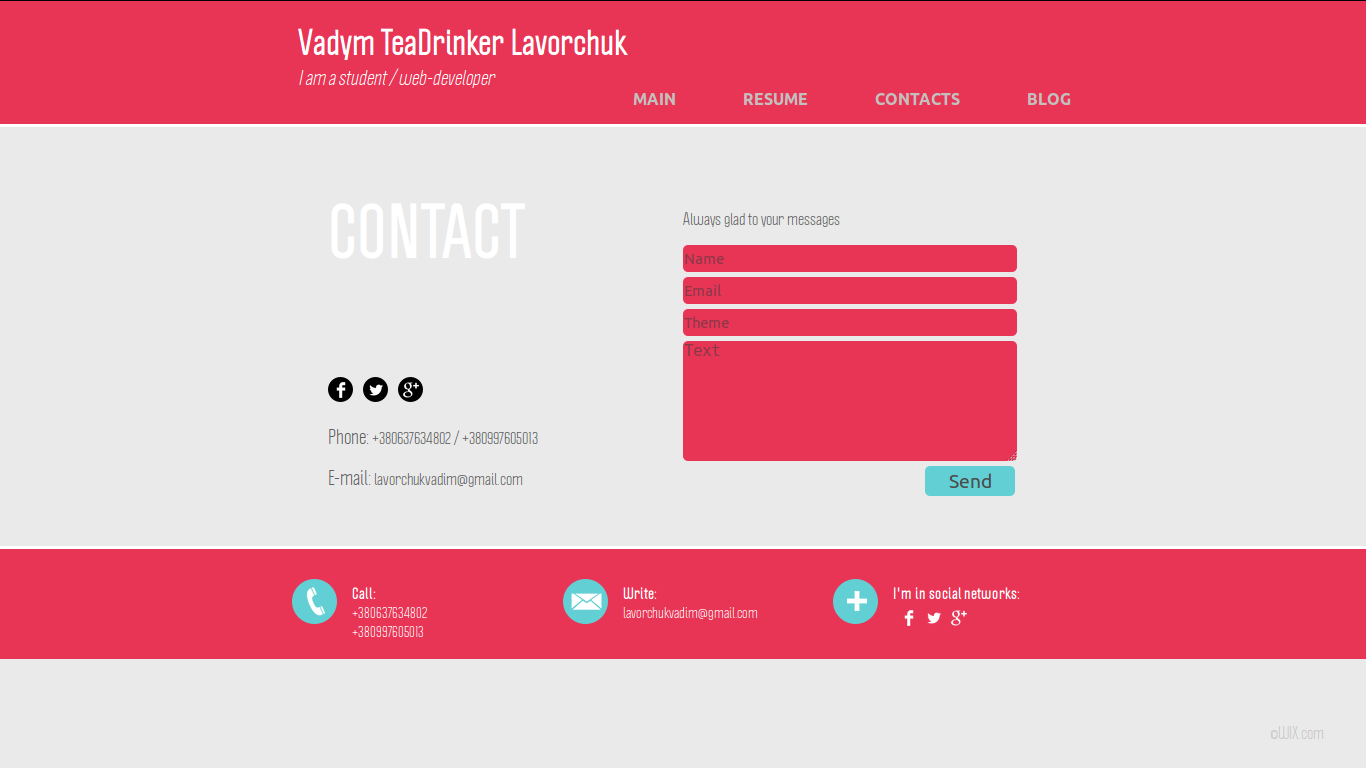
Main



Resume



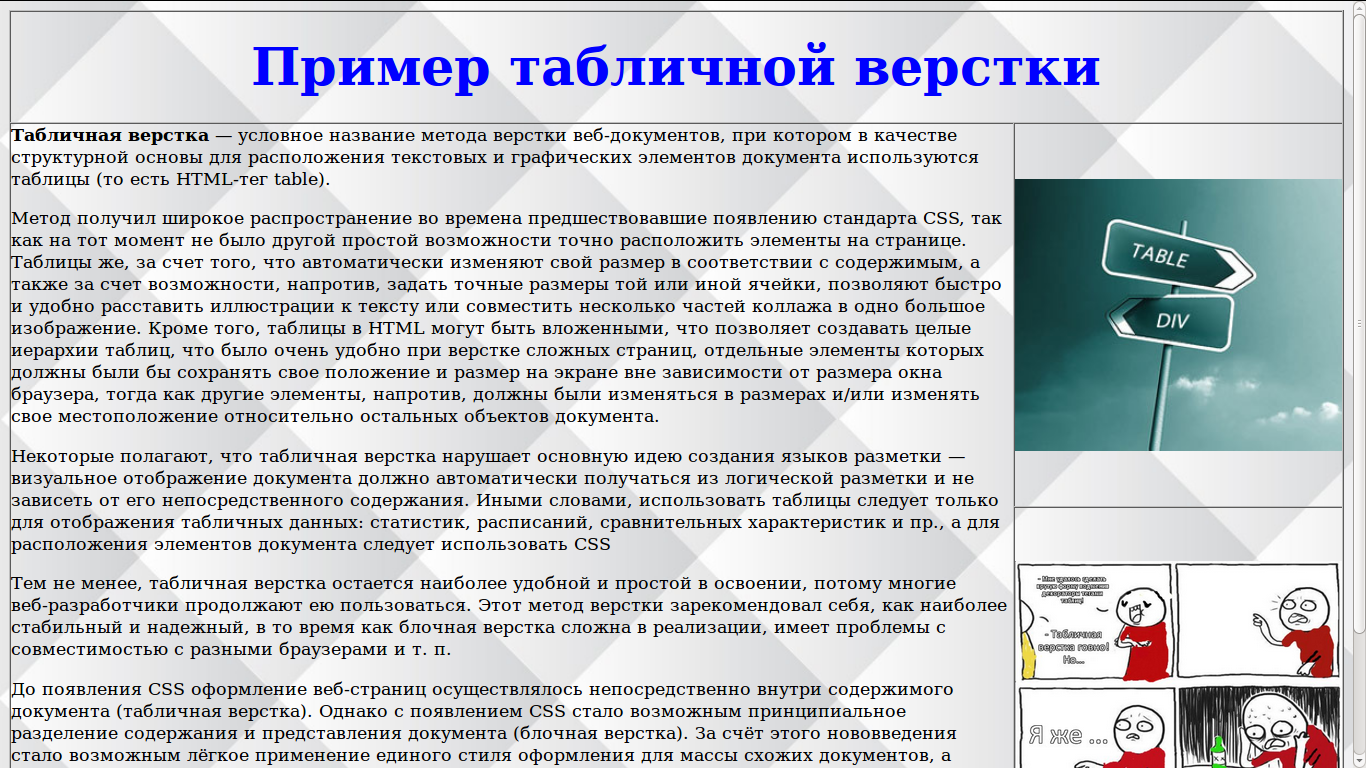
Contacts



Blog



Table layout



**5.3. Меры по обеспечению юсабилити и доступности.**

Концепция универсального «юзабилити» состоит из ряда составляющих. Прежде всего – это доступность, удобство и простота использования, а также универсальность дизайна.

Сайт является общедоступным, без обязательной регистрации. Доступны также все его материалы. Дизайн является универсальным.

Сайт удобен в использовании: навигация имеет не очень много уровней, пользователю легко найти нужную информацию с помощью верхнего меню.

**5.4. Особенности дизайна.**

Многоколоночный тип макета, сверстан div-версткой с приминением CSS. Сайт делиться на такие блоки как: header, content, footer. Header в свою очерень вмещает в себя заголок и навигацию по сайту. Content — информацию в зависимости от выбранной пользователем категории. Footer — дополнительную контактную информацию.

Визуально это типичный информационный сайт. Цвета подобраны удачно, не «режут» глаза. Размер текста средний, удобный для чтения.

**6. Реализация**

**6.1. Клиентская часть. Выбор технологий.**

Используется html5, css3, javascript.

**Index.html**

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8">

<title>TeaDrinker Vadym</title>

<link rel="shortcut icon" href="images/favicon.png" type="image/png">

<link rel="stylesheet" type="text/css" href="styles/main-style.css">

<link rel="stylesheet" type="text/css" href="styles/fonts.css">

</head>

<body>

<div class="wrapper">

<header>

<h1>Vadym <strong>TeaDrinker</strong> Lavorchuk</h1>

<p>I am a student / web-developer</p>

<nav class="main-nav">

<button class="navigation" id="open\_main">Main</button>

<button class="navigation" id="open\_resume">Resume</button>

<button class="navigation" id="open\_contact">Contacts</button>

<button class="navigation" id="open\_blog">Blog</button>

</nav>

</header>

<section class="main" id="main\_cnt">

<div class="i-am">

</div>

<section class="inside">

<h2>Hello!</h2>

<p class="inside-nav">Will be familiar:

<nav class="but-circle">

<button class="inb-nav">Resume</button>

<button class="inb-nav">Tabular layout</button>

<button class="inb-nav">Blog</button>

</nav>

</p>

<p class="some-info">

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec.

</p>

</section>

</section>

<footer>

<nav class="footer-nav">

<div class="phone">

<button></button>

<dl>

<dt>Call:</dt>

<dd>+380637634802</dd>

<dd>+380997605013</dd>

</dl>

</div>

<div class="mail">

<button></button>

<dl>

<dt>Write:</dt>

<dd>lavorchukvadim@gmail.com</dd>

</dl>

</div>

<div class="soc">

<button></button>

<dl>

<dt>I'm in social networks:</dt>

<dd><button class="fb"></button></dd>

<dd><button class="tw"></button></dd>

<dd><button class="g"></button></dd>

</dl>

</div>

</nav>

</footer>

</div>

<script type="text/javascript" src="scripts/main.js"></script>

</body>

</html>

**main-style.css**

@import("reset.css");

body {

font-family: DroidSerif;

color: #606060;

background-color: #eaeaea;

}

body {

margin-bottom: 0px;

}

.wrapper {

margin: 3% 20% 0 20%;

}

header {

background-color: rgba(232, 53, 86, 1);

color: white;

margin: 0 -20.5%;

padding: 0 -88px 10px 290px;

padding-left: 300px;

position: fixed;

top: 0;

width: 100.5%;

padding-bottom: 15px;

border-bottom: 3px solid white;

z-index: 1;

}

header h1 {

font-size: 2em;

}

header p {

margin: -20px 0 0 0;

font-style: italic;

font-size: 1.2em;

}

button {

border: none;

outline: none;

background-color: transparent;

display: inline-block;

transition: 0.3s linear 0s;

}

button:focus {

outline: none;

border: none;

}

.main-nav {

margin: 0 20%;

width: 50%;

}

.main-nav button {

float: left;

font-size: 1em;

text-transform: uppercase;

margin-left: 8%;

font-weight: bold;

color: #c3bfbe;

cursor: pointer;

}

.main-nav button:hover {

color: #61cfd3;

}

.main-nav button:focus {

color: #61cfd3;

}

.i-am {

display: inline-block;

margin: 0 0 0 -5%;

height: 400px;

width: 50%;

border-radius: 50%;

overflow: hidden;

background-image: url("../images/i\_am.jpg");

background-position: 50% 20%;

}

.inside {

display: inline-block;

float: right;

width: 50%

}

.inside h2 {

font-size: 4em;

margin: 0;

}

.inside .inside-nav {

font-size: 3em;

margin: 0;

}

.but-circle {

width: 120%;

margin-left: -5%;

margin-top: 10px;

}

.but-circle button {

height: 130px;

width: 130px;

border-radius: 50%;

color: white;

margin-right: 4%;

text-transform: uppercase;

cursor: pointer;

}

.but-circle button:nth-child(1) {

background-color: rgba(232, 53, 86, 1);

}

.but-circle button:nth-child(1):hover {

background-color: rgba(134, 35, 57, 1);

}

.but-circle button:nth-child(2) {

background-color: rgba(97, 207, 211, 1);

}

.but-circle button:nth-child(2):hover {

background-color: rgba(64, 138, 141, 1);

}

.but-circle button:nth-child(3) {

background-color: rgba(209, 200, 37, 1);

}

.but-circle button:nth-child(3):hover {

background-color: rgba(139, 133, 25, 1);

}

.some-info {

color: #c3bfbe;

font-size: 0.9em;

}

.main {

margin-top: 150px;

display: inline-block;

margin-bottom: 50px;

transition: 0.5s linear 0s;

/\*display: none;\*/

}

button::-moz-focus-inner {

border: none;

padding:0;

margin:0;

}

footer {

display: inline-block;

color: white;

background-color: rgba(232, 53, 86, 1);

margin: 0 -34.5%;

width: 168.8%;

border-top: 3px solid white;

}

.footer-nav {

height: 80px;

margin: 30px 20% 0;

}

.footer-nav div {

float: left;

width: 33%;

}

.footer-nav button {

margin: 0;

height: 45px;

width: 45px;

border-radius: 50%;

background-color: rgba(97, 207, 211, 1);

background-repeat: no-repeat;

background-position: 55% 50%;

cursor: pointer;

}

.phone button {

background-image: url("../images/phone.png");

}

.mail button {

background-image: url("../images/letter.png");

}

.soc > button {

background-image: url("../images/plus.png");

}

.footer-nav div > dl {

display: inline-block;

font-size: 0.9em;

width: 60%;

}

.phone, .mail, .soc {

position: relative;

}

.phone > button, .mail > button, .soc > button {

position: absolute;

top: 0;

left: 20px;

}

.phone > dl, .mail > dl, .soc > dl {

margin-left: 80px;

margin-top: 5px;

}

.footer-nav dt {

font-weight: bold;

}

.footer-nav dd {

margin: 0;

word-wrap: break-word;

}

dd > button.fb, dd > button.tw, dd > button.g {

background-color: transparent;

height: 20px;

margin-left: 5px;

width: 20px;

}

.fb {

background-image: url("../images/facebook.jpeg");

}

.tw {

background-image: url("../images/twitter.png");

}

.g {

background-image: url("../images/google.jpeg");

}

.soc dl dd {

float: left;

}

body:after{

content: "©WIX.com";

color: #c3c3c3;

height: 10px;

left: 93%;

position: fixed;

top: 94%;

width: 50px;

}

.blog {

display: none;

margin-top: 90px;

}

.blog h2 {

text-transform: uppercase;

font-size: 4em;

color: white;

margin-bottom: 50px;

}

.blog .post {

padding: 20px;

width: 60%;

margin-bottom: 20px;

}

.post h3 {

font-size: 2.5em;

margin-top: 0;

}

.post .time {

font-style: italic;

margin-top: -35px;

}

.post img {

float: left;

margin: 0 10px 10px;

width: 65%;

}

.post button.more {

background-color: rgba(97, 207, 211, 1);

border-radius: 5px;

cursor: pointer;

font-size: 1.2em;

height: 30px;

width: 90px;

}

.about-me {

float: right;

width: 30%;

margin-left: 5%;

}

.about-me .im{

background-image: url("../images/i\_am.jpg");

background-position: 50% 20%;

border-radius: 50%;

display: inline-block;

height: 200px;

overflow: hidden;

width: 80%;

}

.about-me p {

padding-bottom: 10px;

border-bottom: 1px solid #c3bfbe;

}

**main.js**

var cnt\_1 = document.getElementById('main\_cnt'),

cnt\_2 = document.getElementById('resume\_cnt'),

cnt\_3 = document.getElementById('contact\_cnt'),

cnt\_4 = document.getElementById('blog\_cnt'),

m\_but\_1 = document.getElementById('open\_main'),

m\_but\_2 = document.getElementById('open\_resume'),

m\_but\_3 = document.getElementById('open\_contact'),

m\_but\_4 = document.getElementById('open\_blog'),

inb\_nav = document.getElementsByClassName('inb-nav');

open\_main.addEventListener('click', function() {

cnt\_1.style.display = "inline-block";

cnt\_2.style.display = "none";

cnt\_3.style.display = "none";

cnt\_4.style.display = "none";

});

open\_resume.addEventListener('click', function() {

cnt\_1.style.display = "none";

cnt\_2.style.display = "inline-block";

cnt\_3.style.display = "none";

cnt\_4.style.display = "none";

});

open\_contact.addEventListener('click', function() {

cnt\_1.style.display = "none";

cnt\_2.style.display = "none";

cnt\_3.style.display = "inline-block";

cnt\_4.style.display = "none";

});

open\_blog.addEventListener('click', function() {

cnt\_1.style.display = "none";

cnt\_2.style.display = "none";

cnt\_3.style.display = "none";

cnt\_4.style.display = "inline-block";

});

**fonts.css**

@font-face {

font-family: "DroidSerif";

src: url("../fonts/Politica.ttf");

font-weight: normal;

font-style: normal;

}

**reset.css**

/\*\*

\* Eric Meyer's Reset CSS v2.0 (http://meyerweb.com/eric/tools/css/reset/)

\* http://cssreset.com

\*/

html, body, div, span, applet, object, iframe,

h1, h2, h3, h4, h5, h6, p, blockquote, pre,

a, abbr, acronym, address, big, cite, code,

del, dfn, em, img, ins, kbd, q, s, samp,

small, strike, strong, sub, sup, tt, var,

b, u, i, center,

dl, dt, dd, ol, ul, li,

fieldset, form, label, legend,

table, caption, tbody, tfoot, thead, tr, th, td,

article, aside, canvas, details, embed,

figure, figcaption, footer, header, hgroup,

menu, nav, output, ruby, section, summary,

time, mark, audio, video {

margin: 0;

padding: 0;

border: 0;

font-size: 100%;

font: inherit;

vertical-align: baseline;

}

/\* HTML5 display-role reset for older browsers \*/

article, aside, details, figcaption, figure,

footer, header, hgroup, menu, nav, section {

display: block;

}

body {

line-height: 1;

}

ol, ul {

list-style: none;

}

blockquote, q {

quotes: none;

}

blockquote:before, blockquote:after,

q:before, q:after {

content: '';

content: none;

}

table {

border-collapse: collapse;

border-spacing: 0;

}

**table.html**

<!-- <!DOCTYPE html> -->

<html>

<head>

<meta charset="utf-8">

<title>Hh</title>

</head>

<body background="images/bg.jpg">

<table border="1" width="100%" height="100%" cellpadding="0" cellspacing="0">

<thead>

<tr>

<th colspan="2" height="15%"><font color="blue" size="42px"><b>Пример табличной верстки</b></font></th>

</tr>

</thead>

<tfoot>

<tr>

<td colspan="2" align="center">Исполнил Лаворчук В.И. TeaDrinker</td>

</tr>

</tfoot>

<tbody>

<tr>

<td rowspan="2" align="left" width="80%">

<p>

<strong>Табличная верстка</strong> — условное название метода верстки веб-документов, при котором в качестве структурной основы для расположения текстовых и графических элементов документа используются таблицы (то есть HTML-тег table).

</p>

<p>

Метод получил широкое распространение во времена предшествовавшие появлению стандарта CSS, так как на тот момент не было другой простой возможности точно расположить элементы на странице. Таблицы же, за счет того, что автоматически изменяют свой размер в соответствии с содержимым, а также за счет возможности, напротив, задать точные размеры той или иной ячейки, позволяют быстро и удобно расставить иллюстрации к тексту или совместить несколько частей коллажа в одно большое изображение. Кроме того, таблицы в HTML могут быть вложенными, что позволяет создавать целые иерархии таблиц, что было очень удобно при верстке сложных страниц, отдельные элементы которых должны были бы сохранять свое положение и размер на экране вне зависимости от размера окна браузера, тогда как другие элементы, напротив, должны были изменяться в размерах и/или изменять свое местоположение относительно остальных объектов документа.

</p>

<p>

Некоторые полагают, что табличная верстка нарушает основную идею создания языков разметки — визуальное отображение документа должно автоматически получаться из логической разметки и не зависеть от его непосредственного содержания. Иными словами, использовать таблицы следует только для отображения табличных данных: статистик, расписаний, сравнительных характеристик и пр., а для расположения элементов документа следует использовать CSS

</p>

<p>

Тем не менее, табличная верстка остается наиболее удобной и простой в освоении, потому многие веб-разработчики продолжают ею пользоваться. Этот метод верстки зарекомендовал себя, как наиболее стабильный и надежный, в то время как блочная верстка сложна в реализации, имеет проблемы с совместимостью с разными браузерами и т. п.

</p>

<p>

До появления CSS оформление веб-страниц осуществлялось непосредственно внутри содержимого документа (табличная верстка). Однако с появлением CSS стало возможным принципиальное разделение содержания и представления документа (блочная верстка). За счёт этого нововведения стало возможным лёгкое применение единого стиля оформления для массы схожих документов, а также быстрое изменение этого оформления.

</p>

<p>

Сразу следует оговорить, что спор вокруг таблиц происходит только в том случае, когда они используются для верстки. Если дело касается хранения табличных данных, то не возникает сомнений, что таблицы именно для этой цели и предназначены.

</p>

</td>

<td align="center" valign="center" height="100%"><img src="images/table1.jpg"></td>

</tr>

<tr>

<td align="center" valign="center" height="100%"><img src="images/table.jpeg" width="100%"></td>

</tr>

</tbody>

</table>

</body>

</html>

**6.1.4. Учет поисковой оптимизации в разметке (проиллюстрировать для какого-либо запроса фрагментами кода)**  
Код описания страницы и ключевые слова:

<meta name="description" content="TeaDrinker, Лаворчук, resume, резюме, Lavorchuk, blog"/>

<meta name="keywords" content="it, it-news, css, c/c++, c#,..." />

Некоторые теги (дескрипторы) языка HTML весьма важны для оптимизации и поискового продвижения сайта. Основные из них: <title>, <h1>… <h6>, <strong>,  <b> ,<i> ,<em>, <img> обязательно должен иметь alt

<img class= «main-image» href= «/images/i\_am.jpg» alt= «TeaDrinker»>

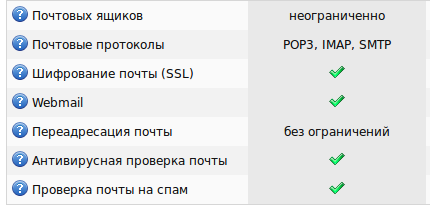
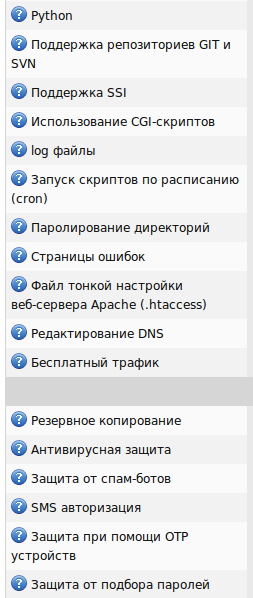
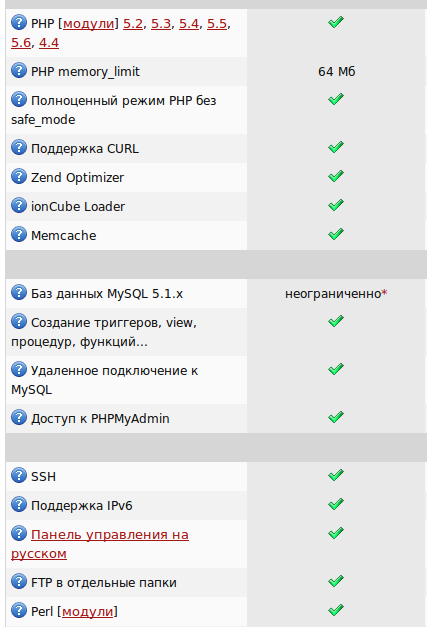
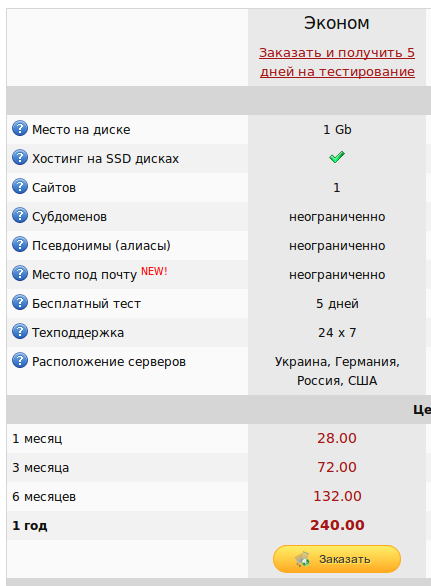
**6.2.Серверная часть – выбор технологий и ПО, среды разработки, CMS, особенности реализации**

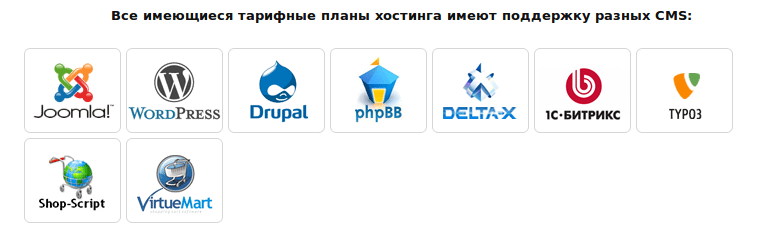
Серверная часть реализована с помощью связки Apache + PHP + MySql. Среда разработки – текстовый редактор SublimeText. Движок – собственной разработки.

**7. Размещение в сети – обосновать выбор способа (собственный сервер / коллокейшн / хостинг), основные требования к серверу/хостингу. Какие возможности хостинга важны для сайта. Выбор домена.**

Для данного проекта выбран хостинг «[ukraine.com.ua](http://www.ukraine.com.ua/)», так как он удовлетворяет все требования проекта данного уровня.

Требования к серверу:





**8.Продвижение ресурса**

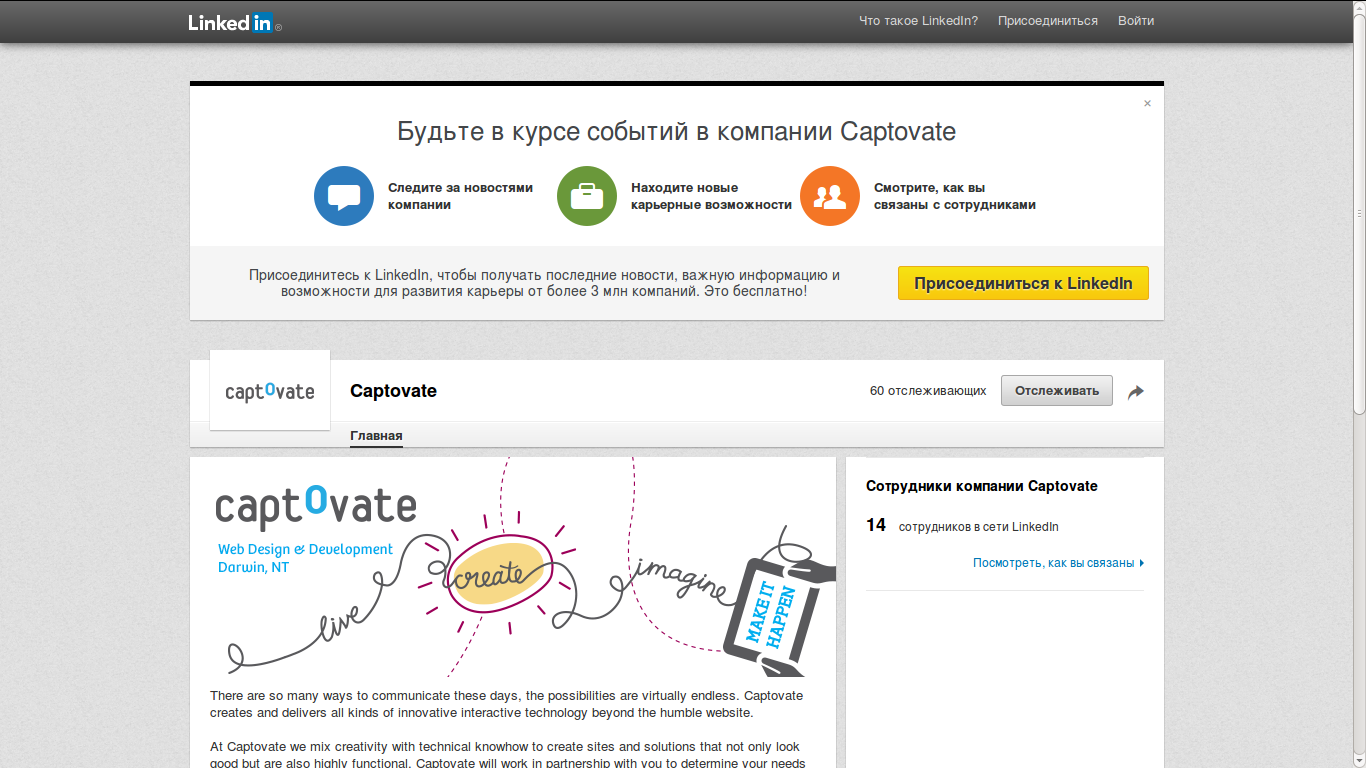
Продвижение сайта имеет такие направления:

1. Необходимость создания на сайте достаточно большого количества качественных и полезных материалов. Так как все материалы пишутся лично мной, я стараюсь писать их на достаточно простом языке с использованием проверочных ресурсов таких как, например <http://api.yandex.ru/speller>, етот ресурс проверяет размещение слов в текстах и орфографию.

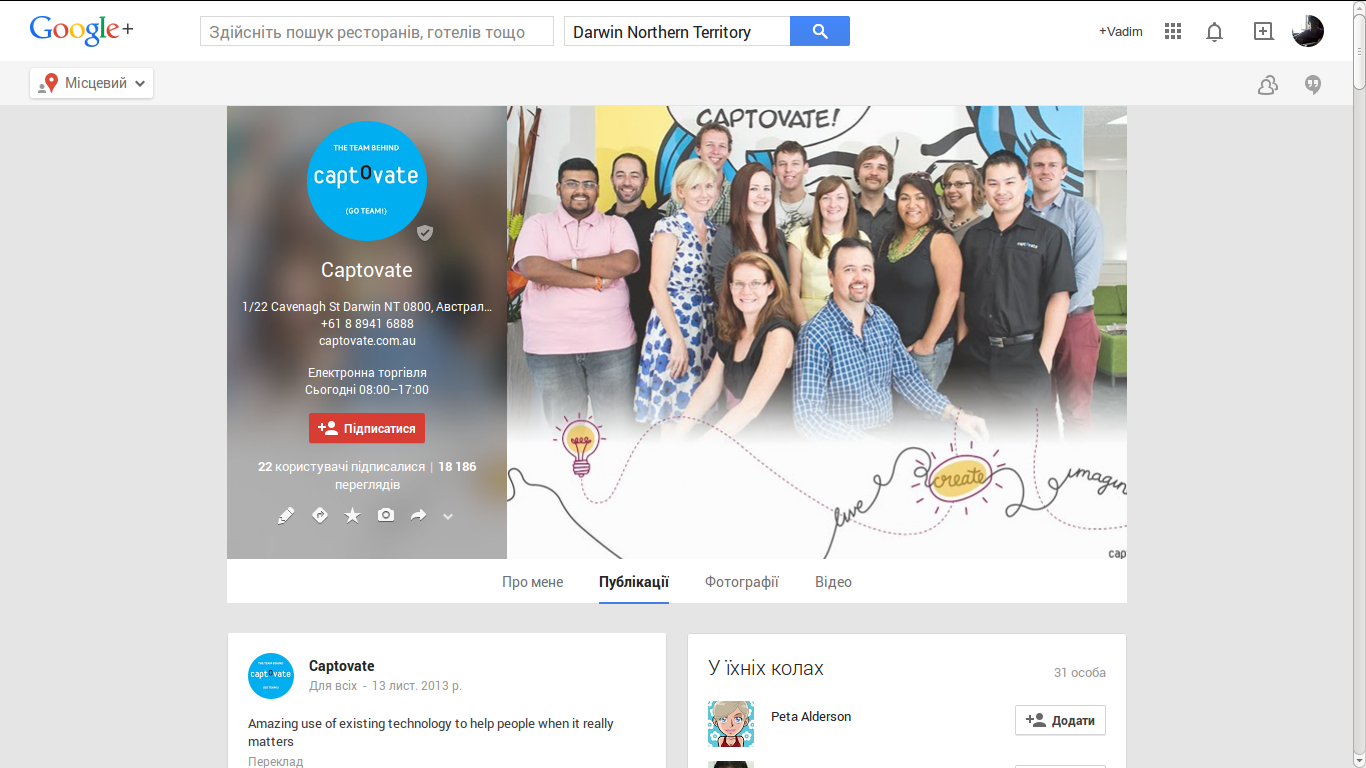
2. Необходимость публикации ссылок на наш сайт (тематически близкие сайты, или сайты клиентов).

Хорошо посещаемый ресурс ето социальные медиа. Любая социальная сеть — ето тысячи пользователей, которым можно рассказать о своем сайте. Пока вы проводите рекламную компанию в соцеальных сетях, списак новых клиентов будет расти до бесконечности.

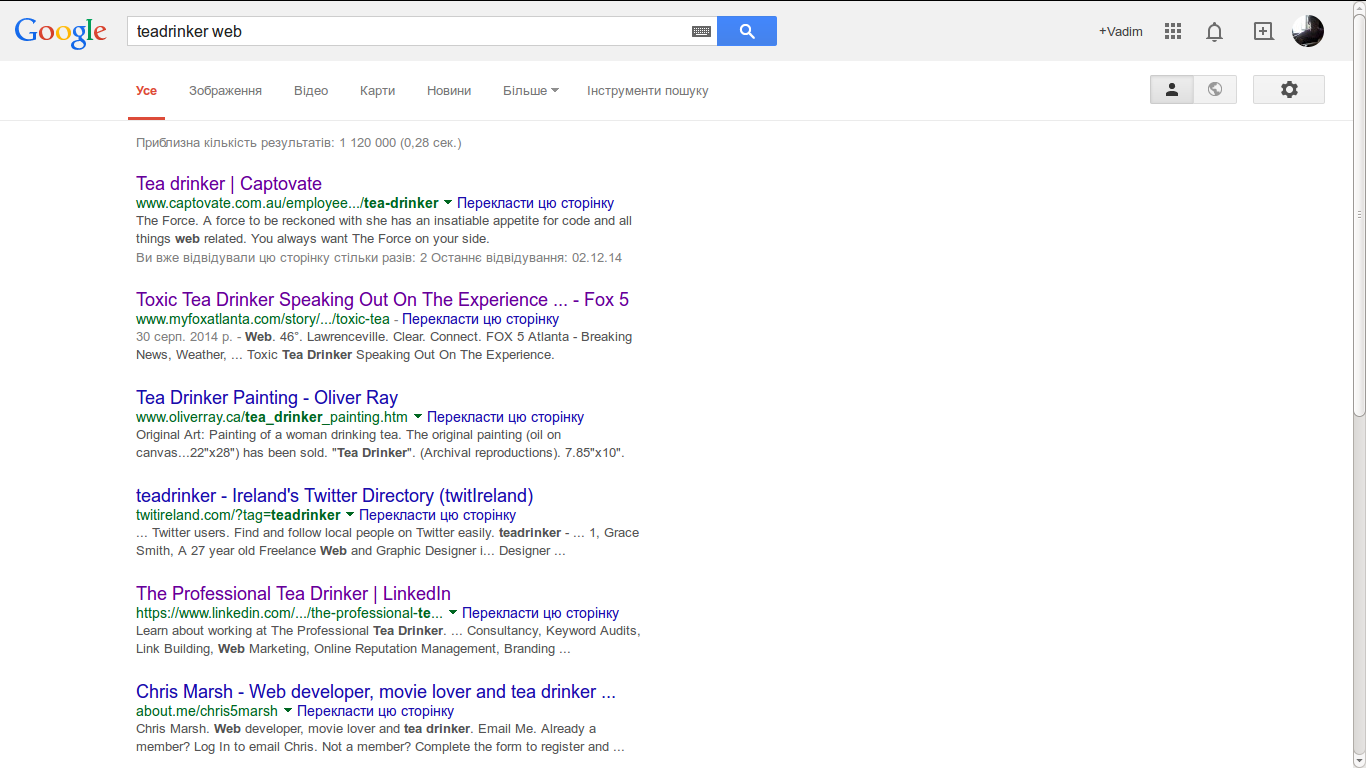
Поетому в социальных сетях созданы группы для компании:







По запросу «teadrinker web» google выдал наш сайт и его ссылку в соц.сеть:



**9. Сопровождение проекта**

В процесс поддержки и сопровождения сайта входит(SEO):

1. регулярное обновление и добавление новой информации (в блоге добавлять актуальную информацию, а в личном разделе обновлять информацию).

2. обеспечение бесперебойной работоспособности сайта и быстрой загрузкой сайта.

3. добавление новых разделов, настройка сайта, розвитие проекта (если мои навыки повышаються и я могу их с пользой использовать в своем проекте, они используются).