Мазмұны

[Кіріспе 3](#_Toc9544882)

[1 Аналитикалық бөлім 4](#_Toc9544883)

[1.1 Пәндік аймақты ұйымдастыру 4](#_Toc9544884)

[1.2 Ақпараттық тапсырмалардың жалпы сипаттамасы 4](#_Toc9544885)

[2 Ақпараттық қамтаманы құру 5](#_Toc9544886)

[2.1 Мәліметтер базасын басқару жүйесін таңдауды негіздеу 5](#_Toc9544887)

[2.2 Мәліметтер базасы кестелерін құру 5](#_Toc9544888)

[2.3 Мәліметтер базасы құрылымын сипаттау 8](#_Toc9544889)

[3 Программалық қамтаманы құру 10](#_Toc9544890)

[3.1 Программалау тілдерін таңдауды негіздеу 10](#_Toc9544891)

[3.2 Қолданылған заманауи технологиялар 11](#_Toc9544892)

[3.3 Жалпы мағлұмат 12](#_Toc9544893)

[3.4 Шақыру және жүктеу 12](#_Toc9544894)

[Қорытынды 22](#_Toc9544895)

[Пайдаланылған әдебиеттер тізімі 23](#_Toc9544896)

Қосымша А

Қосымша Ә

# Кіріспе

Ақпараттық технологиялардың дамуына байланысты күннен-күнге медиа-контент саны арта түсіп жатыр. Себебі, интернеттің дамуы мен технологиялардың жеңілдеуі, әрі арзандауы нәтижесінде кез келген адам аз уақыт ішінде музыка жазу, фильмдер мен мультфильмдер түсіру және т.б. салаларды игеріп жатыр. Ақпараттық технологиялардың арзандауы үлкен компанияларға да көп пайда түсіріп жатыр. Мысалы, ХХ ғасырдың орта кездерінде мультфильмдерді жасау барысында қолданылған 2D анимацияда мультфильмнің әрбір кадрын қолмен салу қажет болса, ғасырдың соңында 3D технологияларының дамуы және арзандауы нәтижесінде мультфильмдерді өндіру жеңілдей түсті. Оған дәлел: 1995 жылы Pixar компаниясымен шығарылған «Toy Story (История игрушек)» толықформатты мультфильм болып табылады. Pixar компаниясының жетістігіне қарап, Disney компаниясы да 3D технологияларды қолдана бастады.

Қазіргі уақытты алсақ, мультсериалдарды санамағанның өзінде, жылына 50-60 данаға жуық мульфильм шығарылып жатыр. Мұндай үлкен ақпараттық ағынның ішінде еркін сезіну үшін әрбір мультфильм мен мультсериал жайлы ақпаратты бір жерде сұрыпталған күйде сақтау қажет.

Бұл практикалық жұмыстың нәтижесінде пайда болатын веб-сайттың басты мақсаты – мультфильмдер мен мультсериалдарға қызығатын кез келген адамға сұрыпталған, толық ақпаратқа қол жеткізуін қамтамасыз ету, және де қолданушыларға олардың талғамы бойынша қызықты мультфильмдер мен мультсериалдарды ұсыну.

Бұл мақсатқа жету үшін келесі іс-әрекеттер жасау керек:

* Деректер базасын басқару жүйесі мен бағдарламалық тілдерді таңдау;
* Қолданылатын заманауи технологиялар таңдау;
* Мультфильмдер мен мультсериалдар жайлы ақпарат көзін іздестіру;
* Деректер базасын құру және жаңа технологиялар қолдана отырып веб-сайт жобалау.

# 1 Аналитикалық бөлім

## 1.1 Пәндік аймақты ұйымдастыру

Пәндік аймақ: Мультфильмдер мен мультсериалдар дереққоры.

Үзінді: Қолданушыларға мультфильмдер мен мультсериалдар іздеу және белгілі мультфильм немесе мультсериал жайлы толық ақпарат ұсынуды қамтамасыз ету.

Пәндік аймақтың сипаттамасы: Ақпараттық технологиялардың дамуына байланысты күннен-күнге медиа-контент саны арта түсіп жатыр. Себебі, интернеттің дамуы мен технологиялардың жеңілдеуі, әрі арзандауы нәтижесінде кез келген адам аз уақыт ішінде музыка жазу, фильмдер мен мультфильмдер түсіру және т.б. салаларды игеріп жатыр. Ақпараттық технологиялардың арзандауы үлкен компанияларға да көп пайда түсіріп жатыр. Мысалы, ХХ ғасырдың орта кездерінде мультфильмдерді жасау барысында қолданылған 2D анимацияда мультфильмнің әрбір кадрын қолмен салу қажет болса, ғасырдың соңында 3D технологияларының дамуы және арзандауы нәтижесінде мультфильмдерді өндіру жеңілдей түсті. Оған дәлел: 1995 жылы Pixar компаниясымен шығарылған «Toy Story (История игрушек)» толықформатты мультфильм болып табылады. Pixar компаниясының жетістігіне қарап, Disney компаниясы да 3D технологияларды қолдана бастады. Қазіргі уақытты алсақ, мультсериалдарды санамағанның өзінде, жылына 50-60 данаға жуық мульфильм шығарылып жатыр. Мұндай үлкен ақпараттық ағынның ішінде еркін сезіну үшін әрбір мультфильм мен мультсериал жайлы ақпаратты бір жерде сұрыпталған күйде сақтау қажет.

Бұл практикалық жұмыстың нәтижесінде пайда болатын веб-сайттың басты мақсаты – мультфильмдер мен мультсериалдарға қызығатын кез келген адамға сұрыпталған, толық ақпаратқа қол жеткізуін қамтамасыз ету, және де қолданушыларға олардың талғамы бойынша қызықты мультфильмдер мен мультсериалдарды ұсыну.

## 1.2 Ақпараттық тапсырмалардың жалпы сипаттамасы

Қарастырылған пәндік аймаққа байланысты ақпараттық жүйеге келесі негізгі міндеттер табылды:

* ақпараттық жүйе Интернет желісіне қосылу мүмкіндігі бар кез келген құрылғыда ашылуы қажет.
* барлық мәліметтер ортақ бөлімдер мен жеке тақырыптарға сұрыптап, жіктелуі керек;
* мультфильмдер мен мультсериалдарда сұрыптау жүргізу және олардың аты бойынша іздеу мүмкіндігі болуы керек;
* ақпараттық жүйе ұсынатын мәліметтің барлығы екі тілде (ағылшынша, орысша) ұсынылуы керек.

# 2 Ақпараттық қамтаманы құру

## 2.1 Мәліметтер базасын басқару жүйесін таңдауды негіздеу

MySQL – дүние жүзінде ең көп қолданылатын, қайнары тегін және ашық, реляцияланған мәліметтер қоры жүйесі (RDBMS). Серверлік бағдарлама ретінде, бірнеше қолданушыларға бірнеше мәліметтер қорын қолдануды қамтамасыз етеді. Толық функционалды мәліметтер қорын қажет ететін, қайнары тегін проекттер жиі MySQL-ды қолданады. Серверді коммерциялық проекттерде қолдану үшін, қосымша функциялармен ұсынылатын бірнеше нұсқалары бар. MySQL-ді қолданатын бағдарламаларға TYPO3, Joomla, WordPress, phpBB, MyBB, Drupal және де басқа LAMP бағдарламаларын жатқызуға болады. MySQL сонымен қатар Google, Wikipedia, Facebook және Twitter сияқты дүниежүзілік және кең ауқымды web-бағдарламаларында қолданылады.

Open Server бағдарламасында MySQL-мен жұмыс істеуге арналған phpMyAdmin веб-интерфейсі бар.

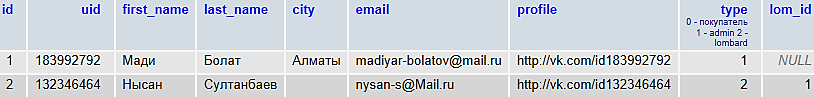
## 2.2 Мәліметтер базасы кестелерін құру

Ең алдымен «lomhub» деректер базасын құрып, ішіне келесі он кестені еңгіземіз. User кестесінде қолданушылардың жалпы ақпараты сақталады. Жаңадан тіркелген қолданушының типі автоматты түрде беріледі. Оны «По умолчанию» бағанынан көре аласыздар. 0-ге тең болса сатып алушы, 1 – админ, 2 – ломбард иесі немесе администраторы.



1 сурет – User кестесі

Мысалы 2-суретте екі қолданушы берілген. Бірінші қолданушының типі администраторға тең, ал екінші қолданушы ломбардтың иесі және оның ломбардының идентификаторы (lom\_id) 1-ге тең.



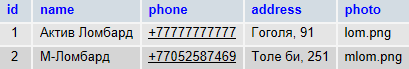
2 сурет – User кестесін толтыру

«Lombard» кестесінде ломбардтар туралы ақпараттар еңгізіледі. Бұл жерде ломбардтың атауы, телефон нөмірі, мекенжайы және фотосуреті (логотип) беріледі. Кейін мекенжай арқылы сайттың ішінде 2GIS картасынан орналасуын көріге болады.



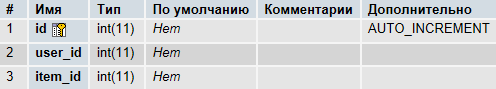
3 сурет – Lombard кестесі

Төмендегі суреттен ломбардтар туралы ақпарат қалай толтырылатынын көруге болады. Мысал ретінде «Актив Ломбард» пен «М-Ломбард» туралы ақпарат еңгізілді. Photo ұяшығына логотип файлының атауы жазылады. Ол файлдар сайттың img директориясында орналасады.



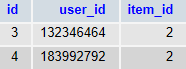
4 сурет – Lombard кестесін толтыру

Веб-сайтта тіркелген қолданушылар сайтқа салынған тауарлардың ішінен ұнағанын өз парақшасында сақтау мүмкіндігі болуы керек. Ол үшін «Bookmark» кестесінде қолданушының идентификаторы мен тауардың идентификаторы еңгізіледі.



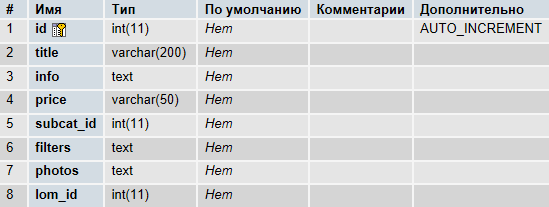
5 сурет – Boookmark кестесі

Мысалы, төмендегі суретке қарайтын болсақ, сайттағы екі қолданушы да идентификаторы 2-ге тең тауарды ұнатқанын білеміз.



6 сурет – Bookmark кестесін толтыру

«Item» кестесінде ломбардтардың тауарлары сақталады. Тауардың аты, сипаттамасы, бағасы, категориясы, фотосуреттері және иесі (ломбард) болады.



7 сурет – Item кестесі

Төменгі суретте сайттағы алғашқы төрт тауар көрсетілген. Бұл тауарлар идентификторлары 1 және 2-ге тең ломбардтарға тиесілі.



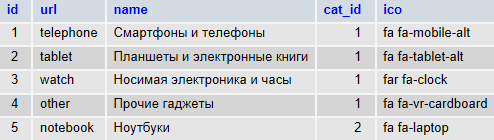
8 сурет – Item кестесін толтыру

«Cat» кестесінде тауарлар категориясы сақталады. Категорияның аты, сілтемесі және белгісі болады. Төменде сайттың барлық категориялары көрсетілген.



9 сурет – Категориялар кестесі

«Subcat» кестесінде категорияға енетін тақырыптар сақталады. Олар тауарды толығырақ сипаттайды. Мысалға компьютерлер категориясында ноутбуктар, дербес компьютерлер, моноблоктар сияқты тақырыптар бар.



10 сурет – Тақырыптар кестесі

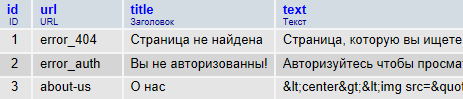
«Filter» кестесінде фильтрлер тізімі бар. Олар тауарларды сұрыптау үшін қажет. Мысалы, смартфондар тақырыбында Apple, Samsung смартфондарын ажырату үшін фильтрлер керек.



11 сурет – Filter кестесі

«Info» кестесінде сайтта қолданылатын мәтіндер және жарнаманың мәтіндері сақталады.

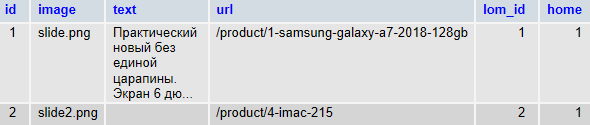
Мәтіннің қысқа сілтемесі, тақырыбы және толық мәтіні болады. Толық мәтіннің ішінде HTML элементтерін қолдануға болады. Бұл текстті ыңғайлы форматтау үшін керек.



12 сурет – Info кестесі

«Poster» кестесінде басты бетте немесе ломбардтың жеке парақшасында көрсетілетін фотослайдттар сақталады. Слайдтың суреті, мәтіні, сілтемесі және иесі (ломбард) болады.

Слайдты ломбард иелері немесе администратор еңгізеді. Ломбард слайдты өзінің жеке парақшасына ғана еңгізе алады. Бірақ администратор оны басты бетке қойғысы келсе ең соңғы, home ұяшығына 1 санын береді. Сол кезде слайд басты бетте де көрсетілетін болады.



13 сурет – Poster кестесі

«Reklama» кестесіне басты бетте көрсетілетін жарнамалар еңгізіледі. Ол жарнамалар «Мы рекомендуем» блокында орналасады және оның тақырыбы, мәтіні, суреті мен сілтемесі болады.



14 сурет – Reklama кестесі

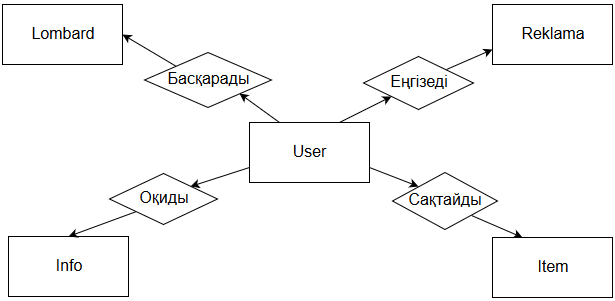
Нәтижесінде пәндік аймаққа қатысы он кесте құрылды. Ол кестелердің барлығы веб-сайтты жобалауда қажет болады.

Ал енді осы деректер базасындағы кестелердің құрылымы мен өзара байланысуын келесі тақырыптан көре аласыз.

## 2.3 Мәліметтер базасы құрылымын сипаттау

### 2.3.1 Нысан-байланыс моделі

Нысан-байланыс моделі (Entity-Relationship model) – бағдарламалық жасақтама инжинрингінде дерекқорды сипаттау үлгісі. Ол пәндік аймақтың концептуалды кескіндерін сипаттауға мүмкіндік беретін деректер моделі. Оның көмегімен негізгі түйіндер мен олардың арасында орнатыла алатын байланыстарды белгілеп алуға болады (15-сурет).



15 сурет – ER модель

ER-модельде пәндік аймаққа қатысты ең басты бес кестенің өзара байланысуы сипатталған.

### 2.3.2 Деректер базасы кестелерінің атрибуттары

Пәндік аймақ пен ақпараттық тапсырмаларды зерттеу нәтижесінде пайда болған атрибуттар тізімі 2-кестеде көрсетілген.

2 кесте Атрибуттар тізімі

|  |  |  |  |
| --- | --- | --- | --- |
| № | Атрибут атауы | Белгіленуі | Типі |
| 1 | Қолданушы идентификаторы | user.uid | int(11) |
| 2 | Қолданушы аты | user.first\_name | varchar(30) |
| 3 | Қолданушы тегі | user.last\_name | varchar(50) |
| 4 | Қолданушы e-mail | user.email | varchar(150) |
| 5 | Қолданушы веб-сайты | user.type | int(11) |
| 6 | Тауардың аты | item.title | varchar(200) |
| 7 | Тауардың сипаттамасы | item.info | text |
| 8 | Тауардың бағасы | item.price | varchar(50) |
| 9 | Ломбардтың идентификаторы | item.lom\_id | int(11) |
| 11 | Ломбардтың телефон нөмірі | lombard.phone | varchar(15) |
| 12 | Ломбардтың мекен-жайы | lombard.address | varchar(150) |
| 13 | Категорияның сілтемесі | cat.url | varchar(50) |
| 14 | Категорияның атауы | cat.name | varchar(150) |

Атрибуттар ER-модельде анықталған кестелерге енеді. Оларды ескере отырып, деректер базасының шектеулері құрылды.

Басты кестелерге қатысты 14 атрибут еңгізілді. Бірақ олардың жалпы саны – 53.

# 3 Программалық қамтаманы құру

## 3.1 Программалау тілдерін таңдауды негіздеу

Бүгінгі уақытта веб-сайт құруға арналған біршама бағдарламалау тілдері бар. Олардың ішінде Python (Django), Ruby (Ruby on rails), Perl (Mojolicious) және де PHP. Python, Perl, Ruby-лардан қарағанда, PHP арнайы веб-сайттар жасауға арналған бағдарламалау тілі. Ал айтылып кеткен үш тіл, бағдарламалаудың басқа да салаларында қолданылады. Сол себептен PHP тілінде жазылған веб-сайттар тезірек жұмыс істейді.

Бұл жұмыста PHP тілін қолдану себептері:

* Wappalyzer компаниясының есептеуі бойынша бүкіл әлемдегі сайттардың 82 пайызы PHP тілін қолданады. Сондықтан веб-сайтты жалғастыру үшін бағдарламашыларды табу қиынға соқпайды.
* Веб-хостингтердің 95 пайызы PHP тілі мен MySQL деректер базасын басқару жүйесін қамтамасыз етеді. Егер сайт басқа тілде жазылған болса, оған хостинг табуға қиын болушы еді.
* PHP тегін бағдарламалық жабдықтама болып табылады.
* Барлық ең танымал операциялық жүйелерімен қолданады.
* Көптеген қосымшалар мен дайын кітапханалар бар.

Javascript – объектілі-бағытталған, имеративті және функционалды стильдерді қамтамасыз ететін мультипарадигмалық бағдарламалау тілі. Javascript бағдарламалаудың көптеген салаларында қолданылады. Веб-бағдарламалауда ол HTML тілі ұсына алмайтын заттарды жүзеге асыру үшін, яғни веб-парақтарға интерактивтілік беру үшін қолданылады.

HTML (HyperText Markup Language) – Ғаламтордағы құжаттарды белгілеу стандартталған тілі. Веб-парақтардың белгіленуі көбінде HTML тілінде сипатталады. HTML тілін компьютер немесе телефон мониторына өңдеп шығаратын бағдарлама браузер деп аталады. HTML парақтар HTTP немесе HTTPS протоколы арқылы браузерға жай немесе шифрленген мәтін түрінде жіберіледі. Бұл жұмыста HTML тілінің ең соңғы, бесінші нұсқасы қолданылады.

CSS (Cascading Style Sheets – каскадтық кестелер стилі) – белгілеу тілі арқылы жазылған құжаттың сыртқы көрінісін сипаттауға арналған формалдық тіл. Көбінесе HTML немесе XHTML белгілеу тілінде жазылған веб-парақтардың сыртқы көрінісін безендіру, сипаттау құралы ретінде қолданылады. Бірақ оған қоса кез келген XML-құжаттарын безендіруде де қолданыла алады. Мысалы, SVG немесе XUL. CSS веб-парақтарды құрушылармен веб-парақтарға түстер, шрифттер, блоктардың орналасуын және басқа да сыртқы көрінсін сипаттау үшін қолданылады.

SQL (Structured Query Language – құрылымдық сұраныстар тілі) – реляциондық деректер базасын басқару жүйесіндегі деректерді басқаруда, құруда және өзгертуде қолданылатын бағдарламалау тілі.

## 3.2 Қолданылған заманауи технологиялар

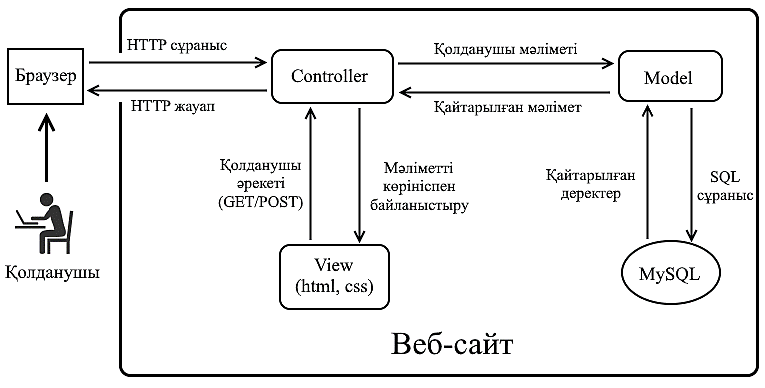
Жиырма бірінші ғасырдың басында, веб-технологиялардың дамуы енді басталғанда, веб-бағдарламашылар PHP сияқты серверлік тілдермен қоса HTML белгілеу тілін бірге қосып жаза беретін. Оған қоса дизайнерлерге сайттың көрінісін өзгертуге қиынға соғатын. Сайтты жасауға жаңадан бағдарламашы келсе, оған қай код не атқаратынын түсіндіру керек болатын. Ал бағдарламаның коды көбейген сайын оның ішінде адасу қиынға соқпайтын.

Қазіргі уақытта осы мәселелерді шешу үшін көптеген шешімдер пайда болды. Солардың бірі объектіге-бағытталған бағдарламалау парадигмасына жататын MVC бағдарламалық жасақтама өңдеу үлгісі болып табылады.

MVC (Model-View-Controller) – бағдарламаның деректерін, қолданушы интерфейсін және басқарушы логикасын, әрқайсысын жеке өзгертуге болатын жеке үш компонентке бөлу схемасы: модель, көрініс және контроллер.

* Модель (Model) деректер ұсынады және өзінің жағдайын өзгерте отырып контроллердің командаларына жауап қайтарады.
* Көрініс (View) модельдің өзгеруіне байланысты қолданушыға модельдің деректерін бейнелеуге жауап береді.
* Контроллер (Controller) модельге өзгеруі қажеттігін жариялай отырып қолданушының әрекетін түсіндіреді.

MVC технологиясы арқылы жасалған веб-сайттың жұмыс істеу схемасы 16-суретте көрсетілген.



16 сурет – MVC жұмыс схемасы

MVC технологиясы арқылы жасалған сайттың артықшылықтары:

* Сайттың логикасы жаңадан келген бағдарламашыларға түсінікті болады.
* Дизайнерлер мен бағдарламашылардың жұмысы екіге бөлініп, екеуі бір-бірінсіз өз жұмысын істей бере алады.

Үлкен көлемдегі код одан әрі түсінікті болып, жаңадан келген бағдарламашылар сайттың кодын тез түсіну үшін объектіге-бағытталған бағдарламалау парадигмасын қолданған жөн.

Объектіге-бағытталған бағдарламалау (Object-oriented programming) – мұрагерлік иерархия құрайтын класстардың ішіндегі белгілі бір класстың данасы болатын объектілер жиынтығы ретінде ұсыну негізінде құрылған бағдарламалау парадигмасы.

API (Application Programming Interface) – бағдарламамен немесе операциялық жүйемен ұсынылған сыртқы бағдарламалық өнімдерде қолдануға арналған дайын класстар, процедуралар, функциялар, құрылымдар және константалар жиынтығы. Веб-бағдарламалауда API-дің Web API деп аталатын түрі қолданылады. Онда деректер XML немесе JSON ретінде қайтарылады. Бұл практикалық жұмыста «TheMovieDB API» қолданылады.

JSON (JavaScript Object Notation) – Javascript негізінде қаланған деректермен алмасу тексттік форматы. Басқа да тексттік форматтар сияқты, JSON жай адамдарға түсінікті болады. XML-ға қарағанда, JSON өте жинақы және қиын құрылымдарды құруда көбірек жақын келеді.

jQuery – Javascript және HTML арасындағы байланысқа негізделген Javascript тіліндегі кітапхана. jQuery кітапханасы кез келген DOM элементіне қол жеткізуге мүмкіндік береді. Оған қоса кітапхана AJAX-пен ыңғайлы жұмыс істеуге арналған API ұсынады. jQuery – ең танымал және ең көп қолданылатын Javascript кітапханасы. 2008 жылы Microsoft және Nokia компаниясы jQuery-мен бірге серіктес болатынын жариялады. Қазіргі уақытта Google, Adobe, Intel, Mozilla, WordPress, IBM және т.б. көптеген компаниялар осы кітапхананы қолданады.

Bootstrap – Twitter жұмыскерлері жасаған сайттар мен веб-бағдарламалар құруға арналған тегін құралдар жиынтығы. Ішінде типографика, веб-формалар, батырмалар, белгілер, кестелер, тізімдер және т.б. веб-интерфейс компоненттерін құруға арналған HTML және CSS безендіру үлгілері бар. Bootstrap – ең көп тараған CSS-фреймворк. Twitter-ден басқа, оны Github, Microsoft, WordPress, Joomla және т.б. компаниялар қолданады.

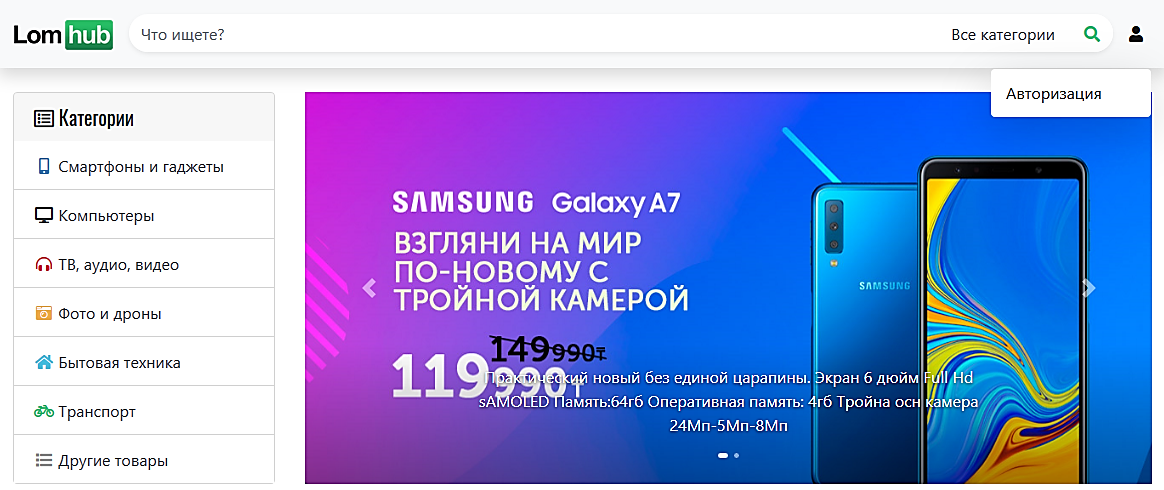
FontAwesome – веб-сайттарда қолданылатын заманауи белгішелері бар қаріптер кітапханасы. Белгішелердің түсін таңдауға, көлемін үлкейтуге және т.б. болады. Бұл белгішелер қарпі ең танымал болып табылады.

## 3.3 Жалпы мағлұмат

Веб-сайт Apache 2.4 веб-серверінде, PHP 5.6 және MySQL 5.6 негізінде жобаланды. Веб-сайтты іске қосу үшін жоғарыдағы модульдерді қамтамасыз ететін кез келген WAMP немесе LAMP пакетін қолдану керек. Мен Open Server 5.2.2 WAMP пакетін қолдандым.

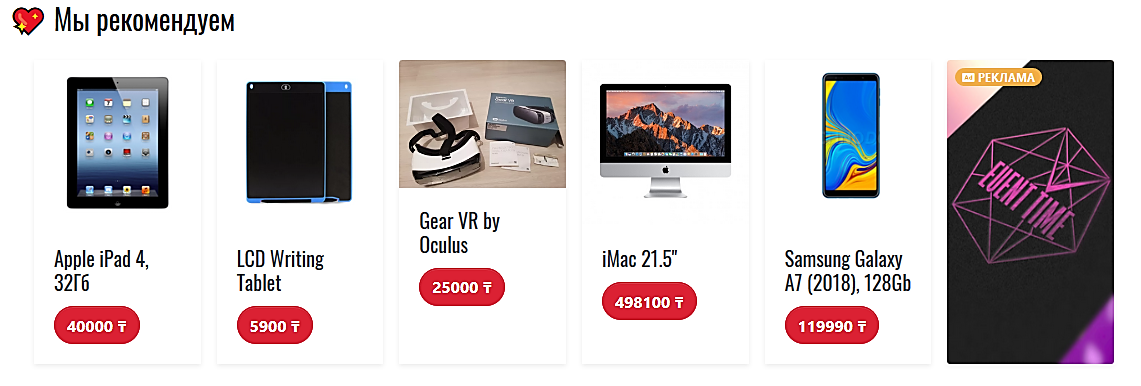
## 3.4 Шақыру және жүктеу

Веб-сайттың жоғарғы бөлігінде логотип, тауарлар іздеу формасы және сайтқа кіру (немесе тіркелу) сілтемелері орналасқан. Басты бетте тауарлар категориясы және жарнама мен басты жаналықтар көрсетуге арналған фотосуреттер слайды (карусель) көрсетіледі.



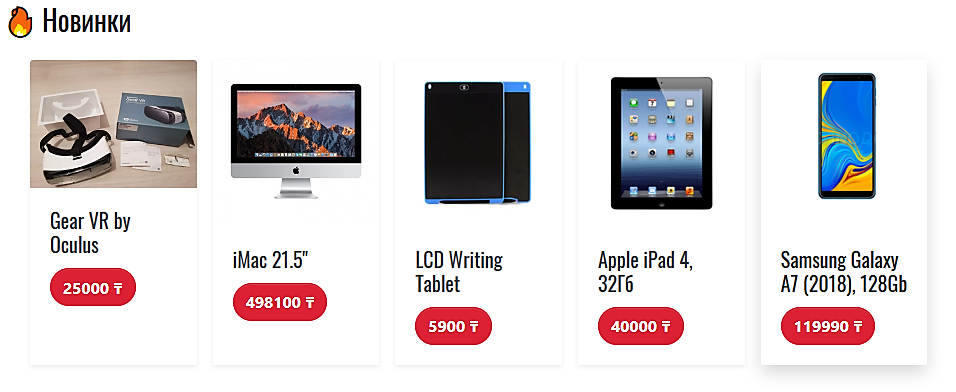
17 сурет – Басты бет

Категориялар мен фотослайдтың астында біздің сайт ұсынатын тауарлар тізімі көрсетіледі. Бұл жердегі тауарлар ломбардтың рейтингі, тауарды көрген адамдар санына және т.б. критерийлер бойынша сұрыпталады. Бес тауардан кейін біздің одақтастарымыздың жарнамасы көрсетіледі. Ол жарнама администратордың рұқсатымен салынады.



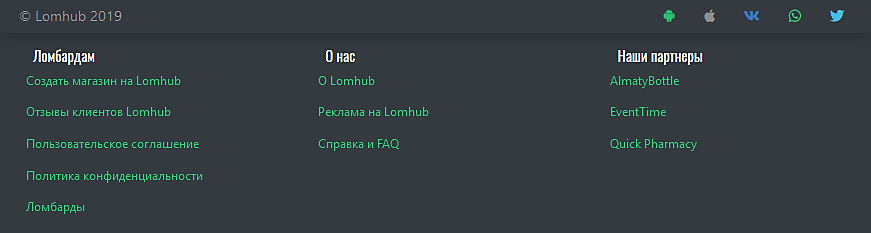
18 сурет – Біздің ұсыныстар

Біздің ұсыныстардан кейін жаңа тауарлар тізімі көрсетіледі. Олар сайтқа еңгізілген уақыт бойынша сұрыпталады. Тауардың аты, бағасы және суреті көрсетіледі.



19 сурет – Жаңа тауарлар

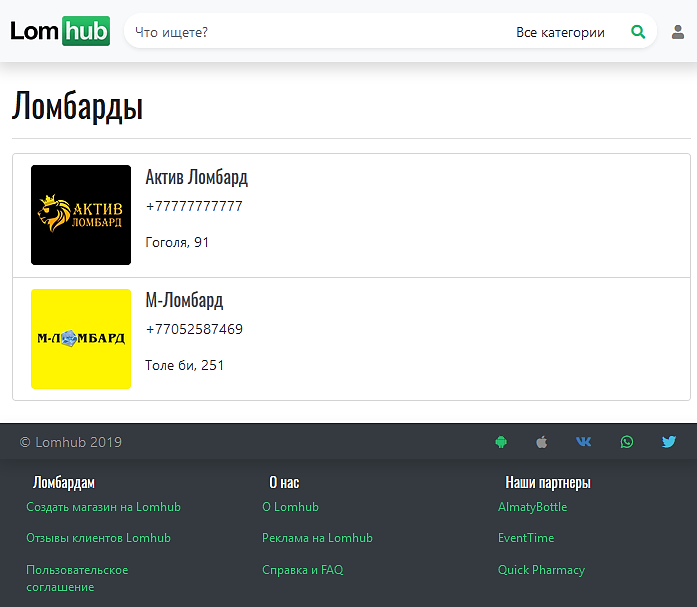
Сайттың ең төменгі жағында сайтқа қатысты мәтіндерге сілтемелер блокы орналасқан. Мұнда сайтқа жаңадан ломбард еңгізу туралы, сатып алушылардың пікірлері, қолданушылық келісім, ломбардтар тізімі, жарнама сатып алу шарттары, серіктестер туралы және т.б. сілтемелер орналасады.



20 сурет – Сайттың төменгі бөлігі

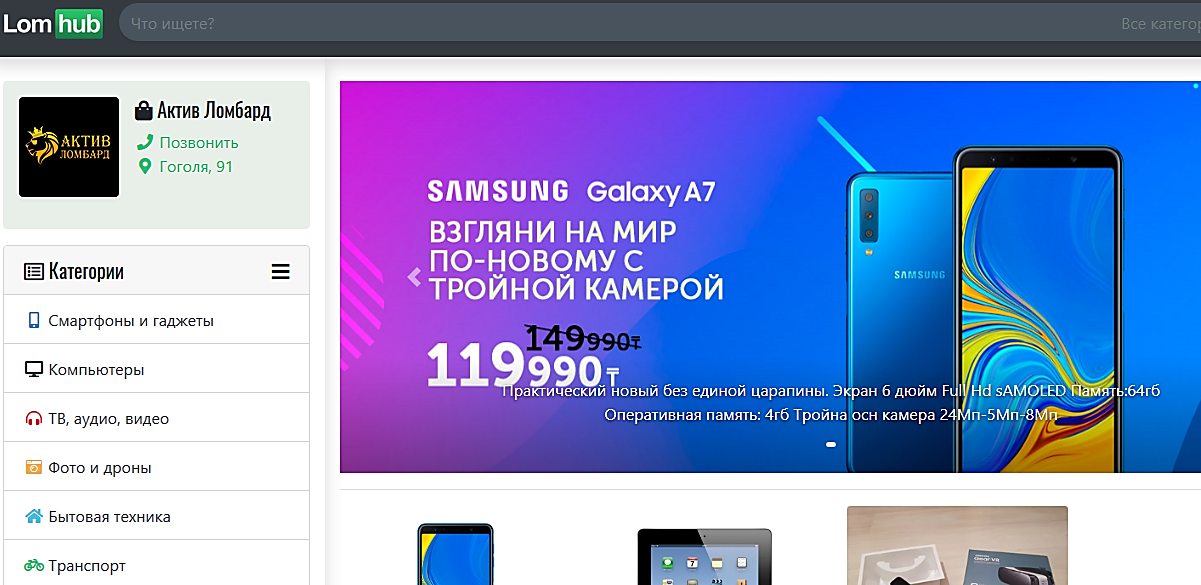
Төменгі жақтан «Ломбарды» сілтемесін басатын болсақ, сайтқа тіркелген ломбардтар тізімі көрсетіледі.

Бұл жерде ломбардтың аты, телефон нөмірі, адресі және суреті көрсетіледі.



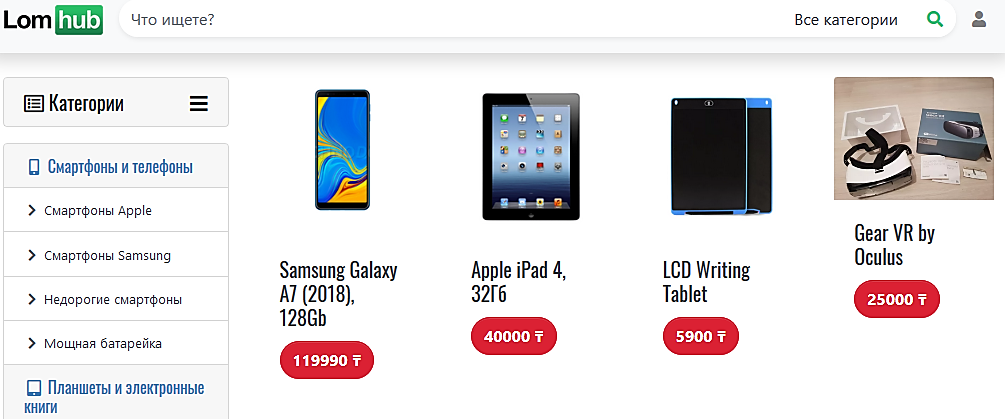
21 сурет – Ломбардтар тізімі

Осы тізімнен ломбардтың атауын бастаны болсақ бізге ломбардтың басты беті ашылады. Ломбардтың парақшасына кірген кезі үстіңгі жолақша қара түске боялады. Бұл бет басты бетке өте ұқсас болады. Категориялар тізімінің үстінде ломбард туралы қысқа ақпарат беріледі.



22 сурет – Ломбардтың парақшасы

Басты беттен бір категорияны, мысалы, «Смартфоны и телефоны» категориясын алайық. Категорияны ашқанда, осы категорияға жататын тақырптар, фильтрлер және тауарлар шығады.



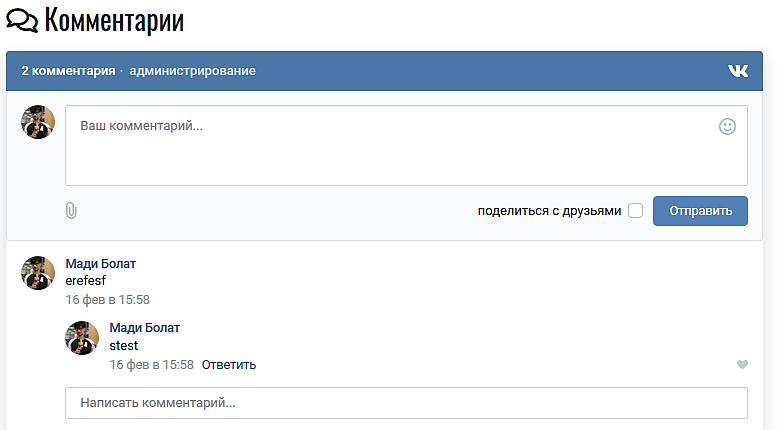
23 сурет – Категориялар парағы

Тауардың парағында фотосуреттер слайдері, жеке парақшаға сақтау батырмасы, тауар туралы ақпарат және иесі (ломбард) туралы ақпарат көрсетіледі.



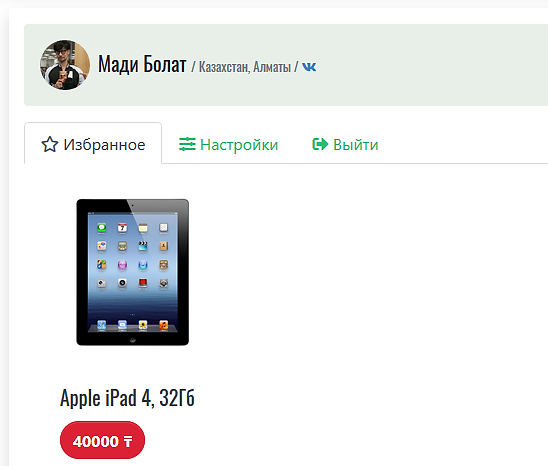
24 сурет – Тауар парағы

Тауардың астыңғы жағында комментарий жазуға арналған ВКонтактенің виджеті орналасқан. Бұл жерде қолданушылар өзінің аккаунты арқылы комментарий жаза алады.



25 сурет – Комментарий жазу

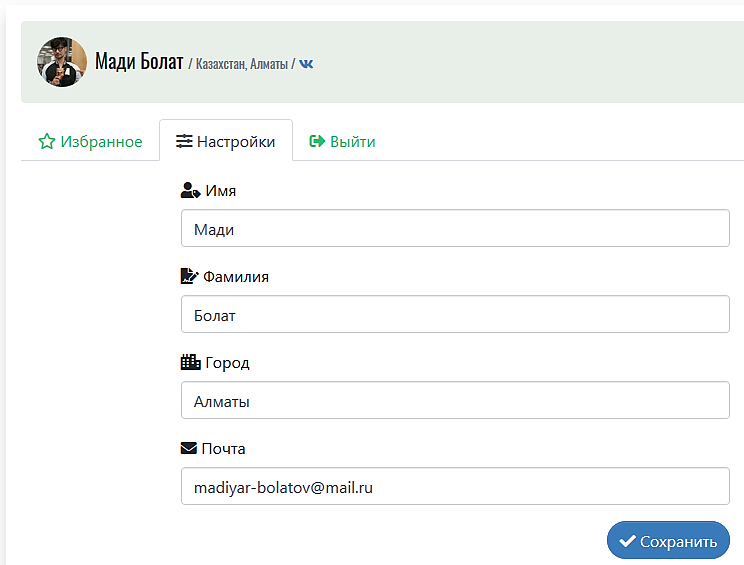
Авторизацияланғаннан кейін қолданушы өзінің парақшасына кіре алады. Бұл жерде ол өзіне ұнаған тауарларды сақтай алады және өзі туралы ақпаратты өзгерте алады. Ол үшін «Настройки» батырмасын басу керек.



26 сурет – Қолданушының жеке парақшасы

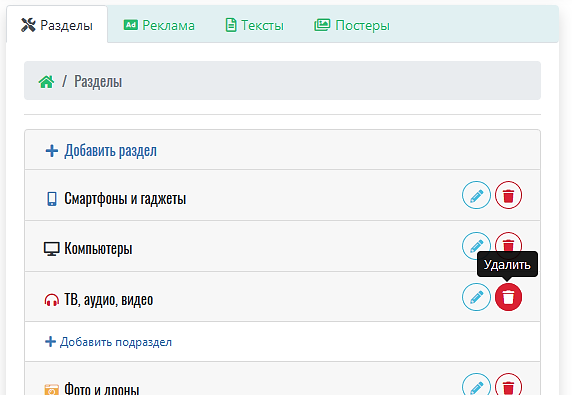
«Настройки» сілтемесін бассақ қолданушының жеке ақпаратын өзгертуге арналған форма ашылады. Бұл жерде қолданушы аты-жөнін, қаласын және электрондық поштасын өзгерте алады.

«Выйти» сілтемесі сайттан шығу үшін арналған.



27 сурет – Қолданушы ақпаратын өзгерту

Егер администратор сайтқа кіретін болса, ол сайтқа жаңадан категориялар, тақырыптар және фильтрлер қоса алады.

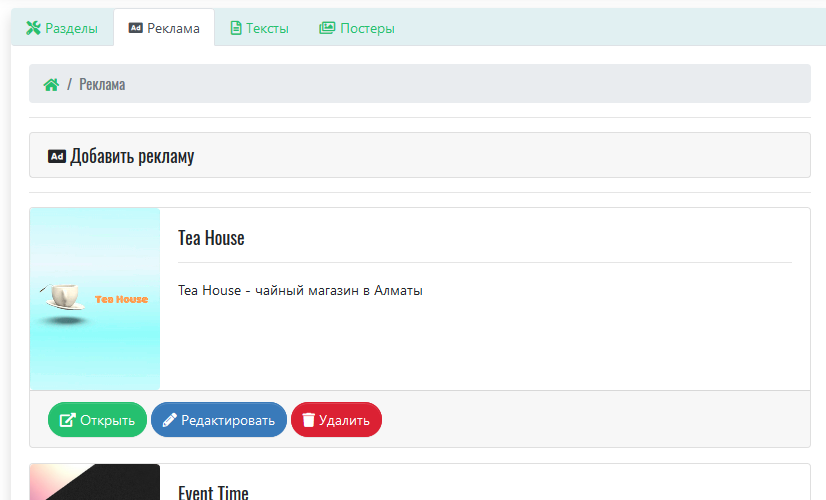


28 сурет – Категориялар парағы

«Реклама» сілтемесін ашатын болсақ, жарнамалар тізімі, оларды өзгерту, жою және жаңадан жарнама қосуға арналған парақ ашылады.

Жаңа жарнама еңгізу үшін «Добавить рекламу» сілтемесін басып жарнама туралы ақпарат: тақырыбы, мәтіні, сілтемесі және суреті еңгізіледі.

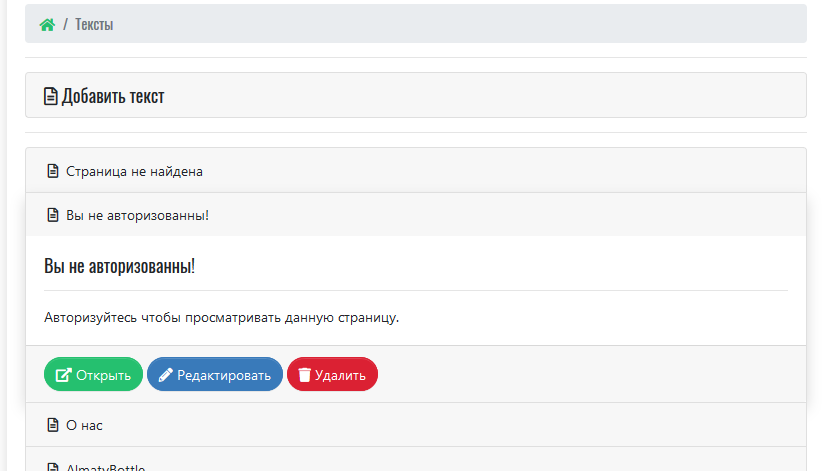
Жарнама басты бетте «Мы рекомендуем» блокында көрсетіледі (18 сурет).



29 сурет – Жарнамалар парақшасы

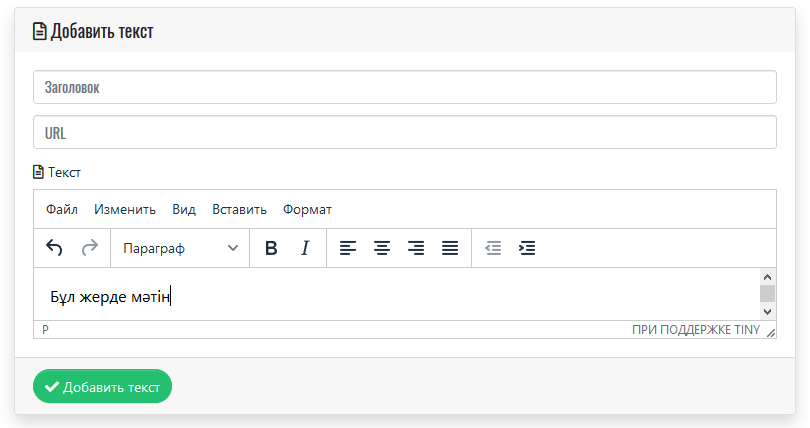
«Тексты» парағында сайтқа қатысты мәтіндерді жоюға, өзгертуге немесе жаңадан мәтін қосуға болады.

Керекті мәтінді өзгерту үшін мәтіндер тізімінен керектісін басу керек. Сол кезі мәтінді өзгерту және жоюға арналған батырмалар пайда болады. Ал егер «Открыть» батырмасын басатын болсақ, мәтін басқа бетте ашылады.



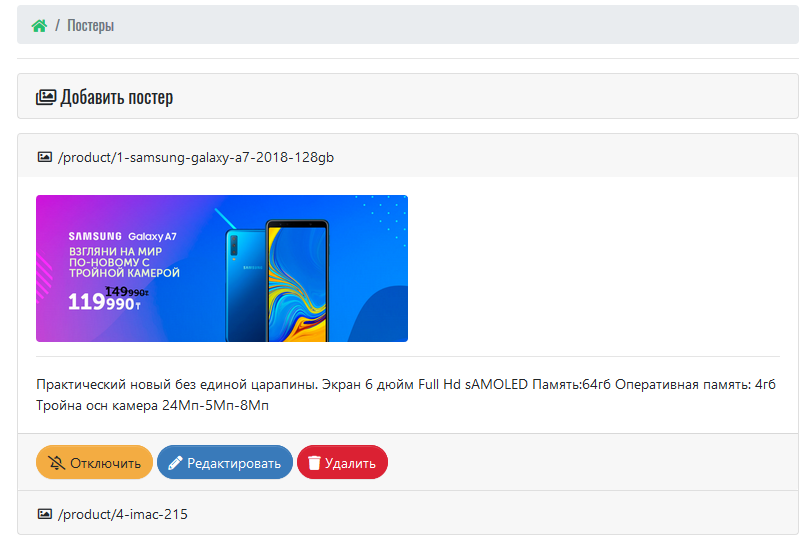
30 сурет – Мәтіндер парағы

«Добавить текст» батырмасын басатын болсақ мәтін еңгізу формасы шығады. Бұл жерде мәтінді форматтау үшін TinyMCE деп аталатын визуалды мәтін редакторы қолданылған. Оның жұмысы MS Word-қа өте ұқсас.



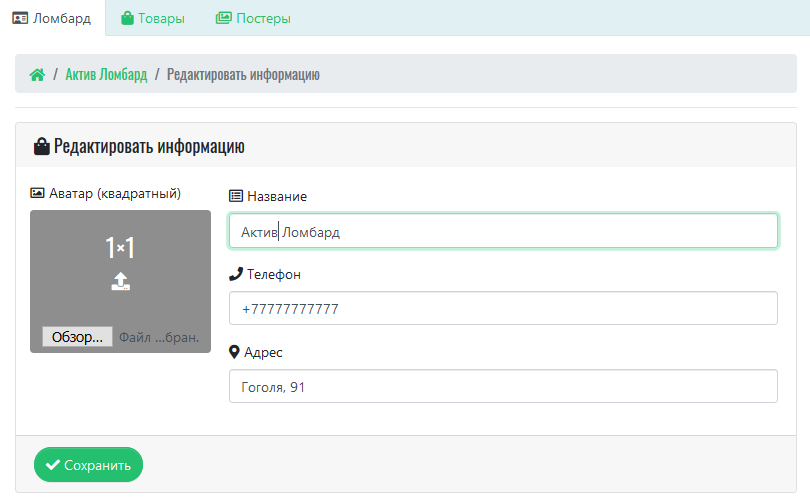
31 сурет – TinyMCE мәтін редакторы

«Постеры» парағында басты беттегі слайдерге постер еңгізуге немесе ол постерлерді өзгертуге, жоюға, сөндіруге болады. Бұл жердегі постерлерді администратор немесе ломбард иесі еңгізеді. Бірақ басты бетке тек қана администратор қоя алады. Ол үшін администратор «Включить» батырмасын басады. Ломбард иесі еңгізген постерлер тек қана ломбардтың жеке парақшасында көрсетіледі.



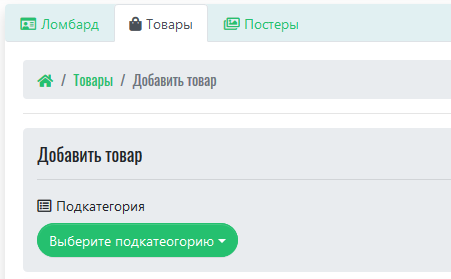
32 сурет – Постерлер парағы

Егер сайтқа ломбард иесі кіретін болса, ол өзінің ломбарды жайлы ақпаратты өзгерте алады, ломбардтың жеке парағына постерлер еңгізе алады және сайтқа жаңа тауар еңгізе алады.



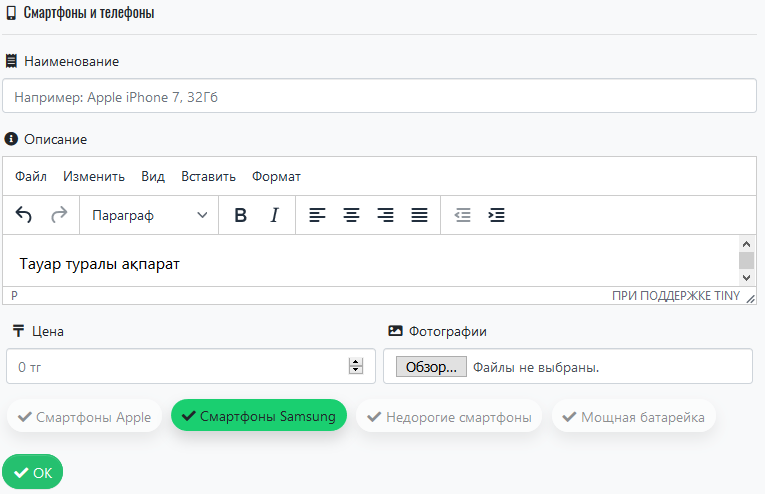
33 сурет – Ломбард жайлы ақпаратты өзгерту

Тауар еңгізу үшін алдымен тауардың категориясын таңдау керек.



34 сурет – Категория таңдау

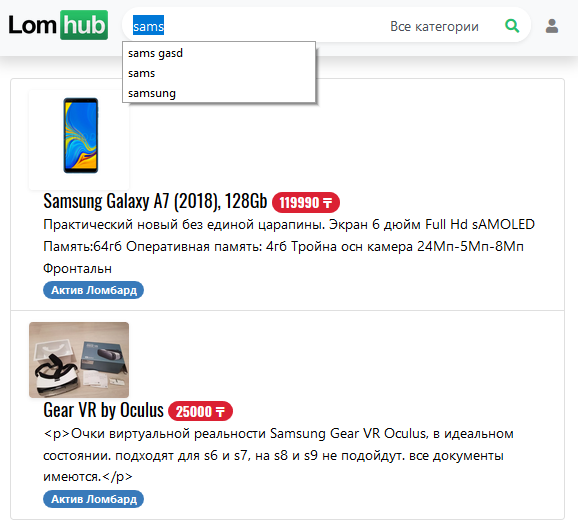
Категория таңдалғаннан кейін тауардың аты, сипаты, бағасы, суреттері және фильтрлер таңдалады. Бұл жерде де TinyMCE қолданылған.



35 сурет – Тауар еңгізу формасы

Сайттың үстіндегі іздеу жолағына тауар атын еңгізу арқылы керекті тауарлар табуға болады.

Мысалы, төмендегі суретте Samsung фирмасының тауарлары көрсетілген. Егер де Samsung фирмасының барлық тауарлары емес, белгілі бір категорияға жататын тауар керек болса, онда ол категорияны «Все категории» тізімін басу арқылы таңдауға болады.



36 сурет – Іздеу нәтижесі

# Қорытынды

Бұл практикалық жұмыстың басты мақсаты – мультфильмдер мен мультсериалдар жайлы толық ақпарат алуға болатын веб-сайт жобалау. Практикалық жұмысты орындау барысында келесі заттар істелінді:

TheMovieDB дерекқоры қолданылды. Нәтижесінде веб-сайтта 20 мыңға жуық м-фильм мен м-сериал бар және олардың саны күніге артып жатыр. Бәріне ортақ, әрі ашық ДБ-сын қолданудың артықшылығы – басқа сайттарға қарағанда ақпараттың көп болуы. Оған қоса бұл ашық (Open Source) ДБ-сын кез келген адам өзгерте алады. Соның арқасында веб-сайтта түзетулер, аудармалар және еңгізулер өз еркімен жүріп жатады. Бұл дерекқормен жұмыс істеу барысында мен API технологиясымен таныстым.

Веб-сайтта ВКонтакте сайтының жаңалықтары көрсетіледі. Бұның басты артықшылығы веб-сайтта жаңалықтар еңгізу үшін арнайы редакторларды жалдап қажеті жоқ. Алдағы уақытта веб-сайтты жобалау жалғастырылатын болса, бұл жаңалықтар арасына басқа да ақпарат көздерін қосуға болады.

Веб-сайт екі тілді етіп жасалды. Сайт екі тілді болуының нәтижесінде сайттың аудиториясы ауқымды болады. Орыс тілін білмеген адам ақпаратты ағылшын тілінде ала алады. Ал ағылшын тілі халықаралық болып саналады.

Мен болашақта осы сайтты аяқтап интернет желісіне қосу мақсатын көздеп отырмын. Сондықтан веб-сайтты жобалауда еңгізілетін функциялар: комментарий және хат алмасу, жаңалық көздерін кеңейту, moonwalk дерекқорымен келісіп веб-сайтта мультфильмдер мен мультсериалдарды тегін, онлайн көру, тематикалық тақырыптарда сөйлесуге арналған форум ашу, қолданушылар рейтингін жасау, смартфондарға клиенттік бағдарлама жасау және т.б.

Бұл сайттың коды <http://github.com/nomadao/cartoon-beta/> парағынан таба алсыз. Сол жерде сайтты өз компьютеріңізге орнату жайлы инструкция бар.

Пайдаланылған әдебиеттер тізімі

1. Васвани В.И. MySQL: использование и администрирование. – М.: Питер, 2011. – 368 с.
2. Воройский Ф.С. Информатика: энциклопедический систематизированный словарь-справочник. – М.: Физмалит, 2006. – 945 с.
3. Самков Г.Б. jQuery. Сборник рецептов. – СПб: БХВ-Петербург, 2010. – 416 с.
4. Скотт Х.А. Администрирование веб-сервера Apache и руководство по электронной коммерции. – М: Вильямс, 2001. – 336 с.
5. Компьютер: электр қауіпсіздігі // Балнұр Сағын. Дата обновления: 13.12.2015. URL: http://el.kz/ (дата обращения: 14.08.2018).
6. Почему именно PHP? // Фрилансер Павел. WiseTIP. [2018—2018]. URL: http://wisetip.ru/page/whyphp (дата обращения: 14.08.2018).
7. Рейтинг языков программирования № 7 // Сообщество программистов DOU. [2005—2018]. URL: https://dou.ua/lenta/articles/language-rating-jan-2016/ (дата обращения: 14.08.2018).
8. Объектно-ориентированное программирование // Википедия. Дата обновления: 24.07.2018. URL: http://ru.wikipedia.org/?oldid=85026496 (дата обращения: 24.07.2018).
9. Apache HTTP Server // Википедия. [2018—2018]. Дата обновления: 24.07.2018. URL: http://ru.wikipedia.org/?oldid=85594509 (дата обращения: 24.07.2018).
10. API // Википедия. [2018—2018]. Дата обновления: 09.08.2018. URL: http://ru.wikipedia.org/?oldid=85888429 (дата обращения: 09.08.2018).
11. August 2011 Web Server Survey // NetCraft. [1995—2018]. Дата обновления: 05.08.2011. URL: https://goo.gl/Y3Kyd6 (дата обращения: 14.08.2018).
12. Bootstrap (фреймворк) // Википедия. [2018—2018]. Дата обновления: 09.08.2018. URL: http://ru.wikipedia.org/?oldid=85878277 (дата обращения: 09.08.2018).
13. CSS // Википедия. [2018—2018]. Дата обновления: 08.08.2018. URL: http://ru.wikipedia.org/?oldid=85868641 (дата обращения: 08.08.2018).
14. HTML // Википедия. [2018—2018]. Дата обновления: 26.07.2018. URL: http://ru.wikipedia.org/?oldid=83906736 (дата обращения: 26.07.2018).
15. JavaScript // Википедия. [2018—2018]. Дата обновления: 06.08.2018. URL: http://ru.wikipedia.org/?oldid=85831928 (дата обращения: 06.08.2018).
16. JSON // Википедия. [2018—2018]. Дата обновления: 06.08.2018. URL: http://ru.wikipedia.org/?oldid=85816531 (дата обращения: 06.08.2018).
17. JQuery // Википедия. [2018—2018]. Дата обновления: 19.08.2018. URL: http://ru.wikipedia.org/?oldid=85504644 (дата обращения: 19.08.2018).
18. Model-View-Controller // Википедия. [2018—2018]. Дата обновления: 04.08.2018. URL: http://ru.wikipedia.org/?oldid=85783223 (дата обращения: 04.08.2018).

#### Қосымша А

#### Веб-сайттың басты PHP коды

<?php

defined('MAD') or die('ERROR');

class Start {

public function run() {

$url = explode('/', strtolower(trim($\_SERVER['REQUEST\_URI'], '')));

unset($url[0]);

// INCLUDE db and core class

require\_once($\_SERVER['DOCUMENT\_ROOT'].'/sys/core.php');

require\_once($\_SERVER['DOCUMENT\_ROOT'].'/sys/func.php');

require\_once($\_SERVER['DOCUMENT\_ROOT'].'/sys/db.php');

// GET class and method from url

if ((empty($url[1]) and !empty($url[2])) or (($url[1] == 'load') and (empty($url[2]) or ($url[2] == 'main')))) {

header('Location: /');

} else {

$class = (!empty($url[1])) ? ucfirst(strtolower($url[1])) : 'Load';

$method = (!empty($url[2])) ? strtolower($url[2]) : 'main';

$default\_method = ($class == 'Load') ? 'text' : 'main';

}

// REQUIRE class and method files

$c\_file = (file\_exists($\_SERVER['DOCUMENT\_ROOT'].'/sys/php/'.$class.'.php')) ? $class : 'Load';

require\_once($\_SERVER['DOCUMENT\_ROOT'].'/sys/php/'.$c\_file.'.php');

// CREATE class and SET vars

if (class\_exists($class)) {

$start = new $class;

} else {

$start = new Load();

$start->err = '404';

}

$start->url = $url;

$start->cat = raw('SELECT \* FROM `cat`');

// RUN method, GEN. path to view's file

if (!empty($start->err)) {

$res = Core::error($start->err);

$view\_file = 'load.error';

} else {

$res = method\_exists($start, $method) ? $start->$method() : $start->$default\_method();

$view\_file = strtolower($class).'.';

$view\_file .= (file\_exists($\_SERVER['DOCUMENT\_ROOT'].'/sys/view/'.$view\_file.$method.'.php')) ? $method : $default\_method;

}

$res['cat'] = $start->cat;

// GEN. VIEW

$this->html($res, $view\_file, $url);

}

public function html($r, $file, $url) {

require\_once($\_SERVER['DOCUMENT\_ROOT'].'/sys/view/.head.php');

require\_once($\_SERVER['DOCUMENT\_ROOT'].'/sys/view/'.$file.'.php');

require\_once($\_SERVER['DOCUMENT\_ROOT'].'/sys/view/.foot.php');

}

}

?>

#### Қосымша Ә

#### Веб-сайттың басты HTML коды

<div class="row">

<aside class="col-lg-3 mt-4">

<div class="card h-100">

<nav class="navbar-expand-lg">

<h5 class="card-header border-bottom-0 clearfix">

<i class="far fa-list-alt"></i> Категории

<a class="navbar-toggler border-0 float-right text-dark" href="#" data-toggle="collapse" data-target="#catCollapse" aria-controls="catCollapse" aria-expanded="false"><i class="fa fa-bars"></i></a>

</h5>

<div class="collapse navbar-collapse" id="catCollapse">

<nav class="list-group-flush w-100">

<? foreach ($r['cat'] as $cat): ?>

<a href="/cat/<?=$cat['url']?>/" class="list-group-item list-group-item-action text-truncate"><i class="fa-fw <?=$cat['ico']?>"></i> <?=$cat['name']?></a>

<? endforeach ?>

</nav>

</div>

</nav>

</div>

</aside>

<div class="col-lg-9 mt-4 d-none d-sm-block">

<div id="homeSlide" class="carousel slide" data-ride="carousel">

<ol class="carousel-indicators">

<? for ($i=0; $i < count($r['poster']); $i++): ?>

<? if ($i==0): ?>

<li data-target="#homeSlide" data-slide-to="<?=$i?>" class="active"></li>

<? else: ?>

<li data-target="#homeSlide" data-slide-to="<?=$i?>"></li>

<? endif ?>

<? endfor ?>

</ol>

<div class="carousel-inner">

<? for ($i=0; $i < count($r['poster']); $i++): ?>

<a href="<?=$r['poster'][$i]['url']?>" class="carousel-item<?=($i==0)?' active':''?>">

<img src="/src/img/poster/<?=$r['poster'][$i]['image']?>" class="d-block h-100" alt="<?=$r['poster'][$i]['image']?>">

<div class="carousel-caption d-none d-md-block mad-caption pb-5">

<p><?=$r['poster'][$i]['text']?></p>

</div>

</a>

<? endfor ?>

</div>

<a class="carousel-control-prev" href="#homeSlide" role="button" data-slide="prev">

<span class="carousel-control-prev-icon" aria-hidden="true"></span>

</a>

<a class="carousel-control-next" href="#homeSlide" role="button" data-slide="next">

<span class="carousel-control-next-icon" aria-hidden="true"></span>

</a>

</div>

</div>

<div class="col-lg-12 mb-4 container-fluid">

<hr><h3 class="p-3">Мы рекомендуем</h3>

<div class="row px-md-5">

<? foreach ($r['best'] as $best): ?>

<div class="card col-xl-2 col-md-4 col-sm-6 p-0 pb-3 px-2 border-0">

<a href="/product/<?=$best['id']?>-<?=urify(translit($best['title']))?>" class="shadow-sm text-dark card-link mad-cards h-100">

<img src="/src/img/product/<?=$best['photo']?>" class="card-img-top" alt="<?=$best['title']?>">

<div class="card-body">

<h5 class="card-title"><?=$best['title']?></h5>

<p class="card-text"><span class="btn btn-danger font-weight-bold"><?=$best['price']?> ₸</span></p>

</div>

</a>

</div>

<? endforeach ?>

<? if (!empty($r['ad'])): ?>

<div class="card col-xl-2 col-md-4 col-sm-6 p-0 pb-3 px-2 border-0 mad-ad">

<a href="<?=$r['ad']['url']?>" class="shadow-sm text-dark card-link mad-cards h-100" target="\_blank">

<span class="badge badge-pill badge-warning m-2 text-white position-absolute"><i class="fa fa-ad"></i> РЕКЛАМА</span>

<img src="/src/img/other/<?=$r['ad']['photo']?>" class="card-img h-100" alt="<?=$r['ad']['title']?>">

<div class="card-img-overlay mx-2 mb-3 rounded text-white">

<h5 class="card-title clearfix"><?=$r['ad']['title']?></h5>

<hr>

<p class="card-text"><?=$r['ad']['info']?></p>

</div>

</a>

</div>

<? endif ?>

</div>

</div>

<div class="col-lg-12 mb-4 container-fluid">

<hr><h3 class="p-3">Новинки</h3>

<div class="row px-md-5">

<? foreach ($r['new'] as $new): ?>

<div class="card col-xl-2 col-md-4 col-sm-6 p-0 pb-3 px-2 border-0">

<a href="/product/<?=$new['id']?>-<?=urify(translit($new['title']))?>" class="shadow-sm text-dark card-link mad-cards h-100">

<img src="/src/img/product/<?=$new['photo']?>" class="card-img-top" alt="<?=$new['title']?>">

<div class="card-body">

<h5 class="card-title"><?=$new['title']?></h5>

<p class="card-text"><span class="btn btn-danger font-weight-bold"><?=$new['price']?> ₸</span></p>

</div>

</a>

</div>

<? endforeach ?>

</div>

</div>

</div>