**ПЛОВДИВСКИ УНИВЕРСИТЕТ**

**„ПАИСИЙ ХИЛЕНДАРСКИ“**

Факултет по Математика и Информатика

Катедра „Компютърна Информатика“

**ДИПЛОМНА РАБОТА**

**Тема:**

Портал за предлагане и търсене на работа.

**Дипломант**: Денислав Найденов Михайлов

**Факултетен номер**: 1101561070

**Специалност**: „Бизнес Информационни Технологии“

**Образователно-** **квалификационна степен**: „Бакалавър“

**Научен ръководител**: гл. Ас. Светослав Енков

**Пловдив 2015**

# Съдържание

Съдържание 2

УВОД 3

Глава 1 6

Водещи кариерни медии 6

Какво е кариерна медия? 6

„The Network“[1] 6

Чуждестранни кариерни сайтове 8

Български кариерни сайтове 10

Глава 2 13

Теоретично решение на проблема, запознаване с технологиите които ще бъдат използвани. 13

Уеб-сървъри 13

HTTP Protocol [6] 14

Apache [7] 14

Статични и динамични уеб сайтове [8] 15

Статични уеб сайтове 15

Динамични уеб сайтове 15

Видове технологии 17

Технологии от страна на потребителя (Front-End Tехнологии) 17

HTML [9] 17

CSS [10] 18

JavaScript [11] 18

jQuery [12] 19

Технологии от страната на сървъра 20

PHP [13] 20

PHP Фреймуъркове [14] 21

MVC Подход 21

База данни [15] 22

MySQL 25

Глава 3 28

Описание на реализацията на системата 28

Видове акаунти 28

Функционални възможности на системата 29

Реализация 31

База данни на системата 31

Front-End Реализация 35

Сорс код (Source Code) на системата 43

Заключение 49

Използвана литература и източници 50

# УВОД

Уеб пространството заема все по-голяма част от нашето ежедневие. Интернет подпомага развитието на младото поколение. Той е вече неизменна част от съвременния живот. Една светла пътека по която ние вървим и без интернет става все по-невъзможно адаптирането на младите хора към съвременния начин на живот. Ние, младите ИТ специалисти, трябва да работим върху това, интернет да се използва от всички млади хора във всички направления на живота, а от по-възрастното поколение, за най-необходимите неща. При по-възрастните да се акцентира върху използването му за черпенето на информация. Те трябва да бъдат обучавани да общуват чрез интернет с по-младите, защото много от семействата вече са извън България и затова интернет е неизменна част от нашия начин на живот.

Когато искаме да се свържем с някой близък човек, използваме портали и програми като Скайп (Skype) и Фейсбук (Facebook), мобилни приложения като Вайбър (Viber) и др. Интернетът е толкова навлязъл в нашето ежедневие, че го използваме за почти всичко. Когато искаме да бъдем информирани за това какво става около нас, ние използваме редица новинарски уеб сайтове, а когато искаме да разберем повече за дадено нещо, ние можем да го потърсим в едни от най-големите търсачки (Google, Yahoo) и до броени секунди ще ни се предостави подробна информация. Най-големият плюс на интернет пространството е това, че е удобно и можем по всяко време да намерим това, което ни интересува.

Вече можем да намерим всичко в уеб пространството, от четенето на последните новини в някой от най-известните български новинарски портали, до гледането на някое видео за развлечение и всичко това ни е достъпно само в разстоянието на един клик. В днешно време съществуват дори портали за намиране на работа, също познати като кариерни медии. С помощта на тези уеб портали, работодатели обявяват своите свободни работни места, а хората които са специалисти в тази област, могат да кандидатстват за тях.

Целта на дипломната работа е разработването на система за предлагане и търсене на работа. Уеб сайтът ще играе ролята на медия, той няма да участва по никакъв начин в избора на кандидати. Ще има единствено ролята да предоставя дадена информация (обяви) на потребителят.

Съществуват 4 типа потребителски акаунти:

* **Нормален потребителски акаунт** – след всяка регистрация потребителят на системата получава нормален потребителски акаунт. Те ще имат възможността да ползват всички функции на сайтът.
* **Модераторски акаунт** – тези акаунти ще имат възможността да редактират обявите и да изтриват обяви.
* **Администраторски акаунт** – хората с администраторски акаунт ще имат всички права както модераторските акаунти, включително и достъп до сорс кодът (Source Code) на сайтът и до администраторския контролен панел.
* **Гости** – Гостите на сайтът ще бъдат хората без акаунт. Те ще бъдат най-ограничени. Ще имат възможността само да разглеждат обявите за работа.

За осъществяването на нашата цел ще имаме нужда от следните работи:

За осъществяването на даденият проект ще ни бъдат необходими следните технологии:

* Създаване на база данни – Тя ще служи за запазване на всякакъв вид информация свързана с потребители, обяви и работодатели. Базата данни ще бъде реализирана чрез MySQL база данни.
* Използване на скриптов език (PHP) за лесна комуникация с базата данни.
* Изграждане на лесен потребителски интерфейс, форми и приятен дизайн с помощта на HTML5 и CSS3.

В настоящата дипломна работа ще бъдат разгледани общите характеристики на този тип информационни системи, както и методите и подходите използвани за изграждането на такъв тип система. Системата ще включва следните функции:

* Добавяне, редактиране и изтриване на обяви.
* Добавяне и изтриване на кандидатури.
* Търсене на обяви.

Текстовата част на дипломната работа се състои от съдържание, увод, три глави, заключение и връзки към използваната литература.

В първата глава ще разгледаме други подобни системи. Ще бъдат разгледани както и най-известните български системи, така и международни.

Във втората глава ще разгледаме подробно основните технологии, които ще ни бъдат необходими за разработването на нашата система.

И в третата глава е посветена на самата система. Тя включва запознанство с концепцията на системата и реализация.

# Глава 1

## Водещи кариерни медии

Какво е кариерна медия?

Нека започнем с въпросът „Какво е кариерна медия“ ? Кариерните медии са уеб сайтове, които се занимават специално с работа или кариера. Много от тези уеб сайтове са проектирани така, че да позволят на работодателите да публикуват обяви за своите свободни позиции. Чрез кариерните сайтове, безработен служител може да потърси работа в своята област, да попълни формуляр или да предостави автобиография по интернет за дадената позиция, след което всички автобиографии се преразглеждат лично от работодателя и имат възможността да бъдат одобрени и извикани на интервю за работа.

„The Network“[1]

The Network е най-голямата международна организация, обединяваща водещите кариерни сайтове в над 100 страни по света. The Network е създадена през 2002 г., когато в състава й влизат едва шест сайта за подбор на персонал. Понастоящем си партнира с 50 компании от 160 държави по целия свят, които притежават общо 36 водещи онлайн трудови борси, оторизирани да работят в 102 страни. В базата данни на The Network има повече от 9.1 милиона CV-та,

които са на разположение на 12.3 млн. регистрирани потребители (работодатели и търсещи работа), имащи достъп до около 700 хил. предложения за работа. Не случайно всеки месец на интернет страницата й се генерира трафик от над 45 милиона уникални посещения (по данни на глобалния лидер в дигиталния маркетинг comScore). Основните предимства за работодателите са, че при търсенето и наемането на персонал не им се налага да се съобразяват с различията във валутите, цените на услугите в отделните страни, дори часовите зони. Бизнесът работи със стандартни договори за наемане на служители (без да се съобразява с изискванията в различните държави), заплаща една и съща цена в една и съща валута (евро), получава еднакво ниво на обслужване по всяко време на денонощието. Срещу съответното заплащане обявите остават в мрежата в продължение на месец. Услугата е най-евтина в Черна гора и Сърбия, където работодателите плащат по 31 евро (в Индия цената е 50 евро, а в Румъния малко по-висока - 69 евро), а най-скъпо е в Швеция (801 евро), Норвегия (793 евро), Франция (690 евро) и Белгия (778 eвро).

Освен това клиентите могат да управляват от едно място всички работни позиции и бюджетите си за тях. Затова експертите в мрежата търсят най-силните играчи (кариерни сайтове) на отделните пазари и държат да работят с тях. По известните й партньори са:   
Yahoo!HotJobs(САЩ),

Totaljobs.com (Великобритания),

StepStone (Германия),

Workopolis.com (Канада),

Cardemploi.fr (Франция),

Kariyer.net (Турция),

HH.ru (Русия),

Naukri.com (Индия)

България също е сред тях и се представлява от онлайн трудовата борса Jobs.bg. В резултат на осъщественото партньорство компании от различни страни публикуват на българския сайт своите свободни позиции, отворени за кандидати от България, а родните фирми получават достъп до партньорската мрежа на The Network и могат бързо и лесно да търсят чуждестранни служители.

Чуждестранни кариерни сайтове

**StepStone - www.stepstone.com**

Сайтът на StepStone е един от основателите на The Network и разполага с една от най-развитите мрежи за подбор на персонал в света. Присъства в много европейски държави (Австрия, Белгия, Германия, Италия, Люксембург, Холандия, Португалия), но най-вече в скандинавските страни (Дания, Норвегия, Швеция). През 2008 г. сайтът е бил използван от над осем хиляди компании от цял свят и разполага с около 40 хил. различни обяви за робота. Характерно за него е, че не използва широкия пазар, а се фокусира върху определени групи – студенти и завършващи обучение, инженери и ИТ специалисти, финансисти и консултанти, фармацевти и медици, HR и маркетинг мениджъри и високи управленски позиции. Занимава се също с предлагане на софтуер за набиране на персонал. Част от клиентите му са известни немски корпорации, като Daimler, Dr.Oetker, BOSCH и Deutsche Telekom.  
  
**Totaljobs, Великобритания - www.totaljobs.uk**  
 Totaljobs.com е водещият кариерен сайт във Великобритания. Има над 1.8 млн. уникални посетители месечно и повече от 150 хил. позиции от около 4000 компании. В базата си данни страницата разполага с 900 хил. CV-та, а всеки месец за работа през него кандидатстват над 800 хил. души. Порталът се стреми да публикува всички свободни позиции, обявени от агенциите по заетостта, затова покрива 30 различни сектора с около 150 хил. незаети работни места. Компанията провежда активен онлайн маркетинг и непрекъснато предлага на клиентите си нови услуги - например, видео към обявите за работа. Организира и виртуални пазари. Много успешна е услугата jobs by email - когато даден работодател публикува обява, срещу цена от 150 евро тя се разпраща на точно 500 кандидати, които напълно отговарят на профила.

**Jobs.ch - www.jobs.ch**  
 Сайтът стартира през 1998 г. и е най-популярният сайт за кариера и образование в Швейцария с над 1.5 млн. посещения месечно. Публикува около 30 хил. работни позиции онлайн, по-голямата част от които са за компютърни специалисти, инженери и офис мениджъри. Доста привлекателна услуга за фирмите е възможността да качат на сайта собственото си лого, сами да поставят рейтинг на обявите си или да ги публикуват в специални категории.  
  
**Adenclassifieds, Франция - www.keljob.com** и **www.cadremploi.fr**  
 Компанията Adenclassifieds е лидер на пазара за подбор на персонал във Франция и работи с два бранда. Keljob.com е сайт за онлайн набиране на персонал, където се публикуват позиции за кандидати с общ профил. Порталът cadremploi.fr пък е специализиран само за високи мениджърски позиции. В архивът му има над 2 млн. автобиографии, а годишно публикува около 100 хил. свободни позиции. Освен да качат обявите си онлайн, работодателите получават достъп и до базата данни със CV-та на всички кандидати за работа. При нужда могат да разчитат дори на съвети и консултации за изграждане имиджа на компанията. Предлага им се също софтуер, които помага на специалистите по човешки ресурси да избират най-подходящите кандидати.  
  
**HeadHunter, Русия - www.hh.ru**  
 Порталът присъства на пазара от 2000 г. и е най-големият сайт за онлайн обяви за работа в Русия. Ежедневно публикува по 40 хил. свободни позиции, а в базата данни от CV-та има над 2.8 млн. резюмета. Освен агенции за подбор на персонал клиенти на HeadHunter са и над 70 хил. компании в цял свят. Порталът е част от Headhunt Group, който разполага с офиси също в Украйна, Казахстан, балтийските републики и Израел.

**Workopolis, Канада - www.workopolis.com**  
Workopolis е най-голямата онлайн трудова борса в Канада и е безспорен лидер на пазара със 75 хил. публикувани работни позиции. Компанията е основана през 2001-а, има офиси в осем града в Канада и става член на The Network през 2008 година. Основният бизнес на Workopolis е публикуването на онлайн обяви и поддържането на база данни от подробни автобиографии. От компанията посочват, че това никак не е лесна задача, защото канадците трудно предоставят лична информация. Затова на сайта има около 5 млн. регистрации, но едва 2 млн. пълни CV-та. Workopolis предлага на клиентите си също банери, възможност за създаване на микросайтове и за провеждане на различни кампании за подбор на персонал.

Български кариерни сайтове

**JOBS.BG – www.jobs.bg** [2]

JOBS.BG е един от най-големите кариерни медии у нас. Той е създаден през 2002 година и от 2003 година са лидери при кариерните медии и генерален източник на кариерна информация за България. Уеб сайтът е посетен от повече от 1.2 милиона уникални посетители, които изпращат повече от 1.8 милиона кандидатури всеки месец. (Google Analytics). JOBS.BG е кариерната медия с най-голяма аудитория у нас, затова повече от 70,000 компании публикуват повече от 25,000 обяви всеки месец.

JOBS.BG се стреми да създаде оптимална среда, даваща възможност на компаниите да открият най-подходящите кандидати, а на кандидатите възможност да реализират пълният си потенциал.

Последвайки големият успех на българският кариерен пазар, JOBS.BG е избран за Business Superbrand за 2007-2008 г., Superbrand за 2009-2010 г. и Superbrand за 2012-2013 г.

**Buljobs – www.buljobs.bg** [3]

Специализираната търсачка за работа Buljobs е собственост на Булджобс ООД - интернет компания специализирана в областта на вертикалното търсене и класификация на информация. Вертикалната търсачка за работа Buljobs предоставя бърз и удобен достъп до обявите в българското уеб пространство като агрегира оферти от три основни типа източници:

* интернет портали за работа,
* фирми-работодатели, които директно търсят служители,
* компании за подбор на персонал.

Чрез специализирана платформа за търсене, Buljobs индексира офертите за работа, следи автоматично за актуалността им и предлага ефективни методи за систематизиране / класифициране и филтриране на съдържанието по различни критерии: профил на позицията, сфера на дейност, населено място, ключови думи, име на фирма или агенция за подбор на персонал.

След избор на оферта от търсачката, потребителят се насочва към първоизточника на обявата (сайта за работа, корпоративния сайт на фирма-работодател или страницата със свободни позиции на HR агенция), от където може да получи подробна информация за вакантната длъжност и за фирмата/ HR агенция които ги предлагат. Търсачката разполага и с двуезичен азбучен класификатор на обявите за работа по професии за съответните бизнес сектори какъвто към момента друг сайт не предлага.

На кандидатите за работа, Buljobs предоставя нов, по бърз и ефективен метод за търсене на актуални оферти за работа в България.

**JobTiger – www.jobtiger.bg** [4]

JobTiger е кариерен уеб сайт и агенция за подбор на персонал, член на Бизнес Група JobTiger. Компанията е създадена през ноември 2000 от Българо-Американския Инвестиционен Фонд със седалище в София, България.

Jobtiger се стремят да предложат интегрирани и ценово ефективни решения, (базирани на Интернет технологиите) при подбора на персонал за своите клиенти, както и продължаващи обучителни модули за повишаване квалификацията. По този начин се стремят да усъвършенстват ефективността на пазара на работната сила в България, за да се подобри и икономическата среда в страната.

Jobtiger предлагат пълен набор от услуги в областта на човешките ресурси– подбор на персонал, обучения, кариерно сътрудничество. Услугите са базирани на Интернет технологиите, което доказано повишава тяхната ефективност и са предназначени за всички физически лица, които търсят промяна в своята професионална кариера, агенции за подбор на персонал, както и компании, предлагащи работни места.

# Глава 2

## Теоретично решение на проблема, запознаване с технологиите които ще бъдат използвани.

За изработването на такъв проект ще ни бъдат необходими няколко вида технологии. На първо място ще ни трябват Front-End технологии, които да направят нашият сайт приятен за окото, лесен за навигация и динамичен за видовете устройства. За целта ще използваме HTML(*HyperText Markup Language*), CSS(*Cascading Style Sheets*).

Също така ще ни бъде необходима база данни, с помощта на която ще запазваме нашата информация, обяви, потребители и др., уеб сървър и някой сървърен език за програмиране с помощта на който ще контактуваме с базата данни - PHP.

Време е да се запознаем с различните видове технологии и термини.

Уеб-сървъри [5]

Уеб сървърът (*Web Server*) е приложна програма, сървър, която позволява на даден компютър да предоставя информация на други компютри, под формата на страници с хипертекст.

Уеб сървърът предоставя информация на клиент под формата на HTML документ. За пренасянето на информацията се използва протокола HTTP. Често Уеб сървърът се използва за предоставяне на информация и в други формати — най често това са изображения във форматите PNG, JPEG и GIF; XML документи и др.

Най-популярните Уеб сървъри са Apache и lighttpd.

**HTTP Protocol [6]**

Протокол за трансфер на хипертекст (*Hypertext Transfer Protocol, съкр. HTTP*) е мрежов протокол, от приложния слой на OSI модела, за пренос на информация в компютърни мрежи. Създаден като средство за публикуване на HTML страници, протоколът довежда до формирането на Световната уеб мрежа. Разработването на протокола е било координирано от Уеб консорциума (*World Wide Web Consortium*) и IETF (*Internet Engineering Task Force*), завършвайки с публикуването на серия от заявления за обсъждане (*RFC*), от които RFC 2616 (от юни 1999) е със статут на стандарт и описва HTTP/1.1 — най-новата и с най-широка употреба HTTP версия.

Apache **[7]**

Apache HTTP Server или само Apache е уеб сървър с отворен код, който има ключова роля за първоначалното разрастване на WWW. Чрез него работят над 70% от сайтовете. Счита се от много специалисти за платформа, според която се разработват и оценяват другите уеб сървъри.

Приложението стартира на много операционни системи, включително Unix, GNU, FreeBSD, Linux, Solaris, Mac OS X, Microsoft Windows, OS/2, Novell NetWare и други платформи.

Apache се разработва от отворено общество от разработчици - Apache Software Foundation. Обществото поддържа и разработва множество други проекти най-важните от които са Apache Ant, Apache SpamAssassin, Apache Tomcat и огромен брой проекти основани на XML езика, но Апачи е най-известния и широко разпространен продукт.

Статични и динамични уеб сайтове **[8]**

Статични уеб сайтове

До преди няколко години беше масова практика да се изработват статични уеб сайтове. Специфичното за тези сайтове е, че няма администрация, от където собственикът на сайта може да променя съдържанието му.

Статичният сайт се състои от електронни страници, реализиране чрез HTML (Hyper Text Markup Language) и CSS (Cascading Style Sheets). В някои случаи се добавя и JAVA или Java Script технология. При избора на статичен сайт е важно да сте предвидили малко на брой страници, в които да поместите информацията за фирмата си, защото поддръжката на сайта може да се извършва само от специалист дизайнер, и това генерира допълнителни разходи за вас. Ако сайтът ви ще съдържа голямо количество информация, която ще се променя често, е по-добре да поръчате динамичен уеб сайт.

Предимствата на статичния сайт са пълна свобода при реализиране на дизайна на сайта, по-бързо зареждане на страниците в браузъра и по-добра СЕО успеваемост.

За съжаление, изборът на статичен сайт ограничава собственикът му в актуализирането на информация по сайта. За промяната дори на една буква, е необходима намеса на дизайнер или програмист! Това само по себе си не е рентабилно за клиента, защото генерира постоянно допълнителни разходи.

Динамични уеб сайтове

Нека поговорим малко и за динамичните уеб сайтове, защото все пак, нашата система ще бъде един динамичен уеб сайт.

Динамичните уеб сайтове разполагат със система за управление на съдържанието (*CMS*), която позволява хора без специфични технически познания, да променят в реално време динамичното съдържание на уеб сайта. Всяка административна система разполага с *WYSIWYG* редактор, съдържащ интерфейс и функции, подобни на познатата на всеки потребител програма за текстообработка *MS Word*.

С няколко кликвания всеки може да добави текст, снимка, хипервръзка, таблица!

Освен възможността за промяна на съдържанието, динамичният уеб сайт има и друго важно предимство - позволява имплементиране в сайта на повече функционалности, необходими на по-големите сайтове - филтриране на продукти, детайлно търсене и сортиране на база определен признак и пр.

При реализиране на дизайн за динамичен сайт, концепцията следва да се съобрази с функционалността на системата за управление на съдържанието, което до някъде е ограничаващо.

Като недостатък на динамичния сайт се приема по-бавното зареждане на страниците и по-трудната *СЕО* обработка на сайта, НО! От както започнаха масово да се изработват динамични сайтове, се наложи като стандарт страницата да се отваря в рамките на 1 секунда! В противен случай сайтът става неизползваем за потребителите.

По отношение на *СЕО* на динамичен сайт, също като правило, се наложи реализиране на *CMS*-системи, които съдържат модул за така наречената *On Site* оптимизация, която позволява дори без помощ от специалист по оптимизация, да се въведат през администрация подходящи ключови думи и мета описание, да се генерират *Simple Links* и заглавия на страниците.

В динамичните сайтове могат да се интегрират различни модули, които Вие да администрирате:

*Модул "Новини"*

*Модул "Галерия"*

*Модул "Портфолио"*

*Модул "Често задавани въпроси"*

*Модул „Обратна връзка"*

*Модул „Кариери"*

*Модул „Абонамент за новини"*

*Модул "Продуктов каталог"* *и др*.

Видове технологии

Технологии от страна на потребителя (Front-End Tехнологии)

HTML **[9]**

HTML (*HyperText Markup Language*) е основният маркиращ език за описание и дизайн на уеб страници. HTML е стандарт в Интернет, а правилата се определят от международния консорциум *W3C*.

Текущата версия на стандарта е HTML 5.0 (*от 28 октомври 2014 г.*)

Описанието на документа става чрез специални елементи, наречени HTML елементи или маркери, които се състоят от етикети или тагове (*HTML tags*) и ъглови скоби (*Пример:* ***<html>***). HTML елементите са основната градивна единица на уеб страниците. Чрез тях се оформят отделните части от текста на една уеб страница, като заглавия, цитати, раздели, хипертекстови препратки и т.н. Най-често HTML елементите са групирани по двойки. (*Пример:* ***<h1> </h1>***).

В повечето случаи HTML кодът е написан в текстови файлове и се хоства на сървъри, свързани към Интернет. Тези файлове съдържат текстово съдържание с маркери – инструкции за браузъра за това как да се показва текстът. Предназначението на уеб браузърите е да могат да прочетат HTML документите и да ги превърнат в уеб страници. Браузърите не показват HTML таговете, а ги използват, за да интерпретират съдържанието на страницата.

Основното предимство на HTML е, че документите, оформени по този начин, могат да се разглеждат на различни устройства, а не само на екрана. Документът може да бъде правилно оформен и върху монитора на персонален компютър, и върху миниатюрния дисплей на пейджър или мобилен телефон.

CSS **[10]**

CSS (*Cascading Style Sheets*) е език за описание на стилове (*Style Sheet Language*). Използва се основно за описване на представянето на документ, написан на език за маркиране. Най-често се използва заедно с HTML, но може да се приложи върху произволен XML документ. Официално спецификацията на CSS се поддържа от W3C.

CSS е създаден с цел да бъдат разделени съдържанието и структурата на уеб страниците отделно от тяхното визуално представяне. Преди стандартите за CSS, установени от W3C през 1995 г., съдържанието на сайтовете и стила на техния дизайн са писани в една и съща HTML страницата. В резултат на това HTML кода се превръща в сложен и нечетлив, а всяка промяна в проекта на даден сайт изисквала корекцията да бъде нанасяна в целия сайт страница по страница. Използвайки CSS, настройките за форматиране могат да бъдат поставени в един единствен файл, и тогава промяната ще бъде отразена едновременно на всички страници, които използват този CSS файл.

JavaScript **[11]**

JavaScript е програмен език, който позволява динамична промяна на поведението на браузъра в рамките на дадена HTML страницата. JavaScript се зарежда, интерпретира и изпълнява от уеб браузъра, който му осигурява достъп до Обектния модел на браузъра. JavaScript функции могат да се свържат със събития на страницата (*Пример: движение/натискане на мишката, клавиатурата или елемент от страницата, и други потребителски действия*). Javascript е най-широко разпространеният език за програмиране в Интернет. Прието е JavaScript програмите да се наричат скриптове.

JavaScript може да влияе на почти всяка част от браузъра. Браузъра изпълнява JavaScript кода в цикъла на събития т.е. като резултат от действия на потребителя или събития в браузъра (*Пример:* ***document.onLoad***).

Основни задачи в повечето JavaScript приложения са:

* Зареждане на данни чрез AJAX
* Ефекти с изображения и HTML елементи: скриване/показване, пренареждане, влачене, слайд шоу, анимация и много други.
* Управление на прозорци и рамки.
* Разпознаване на възможностите на браузъра.
* Използване на камерата и микрофона.
* Създаване на 3D графики WebGL.
* По-добър и гъвкав потребителски интерфейс

Какво не може да се прави с помощта на JavaScript:

* Не може да се записва информация на потребителския компютър или отдалечения сървър.
* Не може да се запазва информация директно в отдалечена база данни.
* Не може да се стартират локални приложения.

jQuery **[12]**

jQuery е ***JavaScript*** библиотека, публикувана в началото на 2006 от Джон Резиг. В основата си jQuery опростява достъпа до всеки елемент на дадена уеб-страница, като по този начин позволява лесно изграждане на динамична функционалност в страниците.

jQuery е безплатен и open source софтуер, лицензиран под MIT лиценз. jQuery се използва в 55% от 10000-те най-посещавани сайтове, което я прави най-популярната JavaScript библиотека днес.

Технологии от страната на сървъра

PHP [13]

PHP е скриптов език със синтаксис базиран на *C* и *Perl*. Използва се предимно в Web-среда за реализиране на широк кръг от услуги. Той е един от най-популярните езици за програмиране в Интернет и популярността му расте непрекъснато.

PHP се разпространява под отворен лиценз (*PHP License*), който по своята същност е *BSD* лиценза и който позволява безплатно разпространяване на програмния код на интерпретатора на езика, както и създаването на производни интерпретатори под други лицензи с уговорката, че тези интерпретатори не могат да включват PHP в името си. Фактът, че PHP се разпространява свободно, го прави удачен избор за изграждане на Web-сървър базиран изцяло на свободни продукти - *GNU/Linux*, *Apache*, *MySQL*/*PostgreSQL* и др.

При поискване, кодът, който е написан на PHP се интерпретира от уеб сървъра на който е качен, и резултатът се връща на уеб браузърът. Потребителят не може да види чистият PHP код без да има достъп до самият файл в който той е записан. По този начин е помислено за сигурността. PHP файловете могат да съдържат *Tекст*, *HTML*, *CSS*, *JavaScript* и *PHP код*. PHP файловете имат разширение .php.

Самият език е преносим на много изчислителни архитектури и операционни системи като *GNU/Linux, UNIX, Mac OS X, Windows*.

Съществуват множество модули (*разширения*) за PHP, които добавят различни функционалности и позволяват много по-бързо и ефективно разработване.

PHP може да работи с повечето модерни бази данни - *MySQL, PostgreSQL, Microsoft SQL Server, Oracle, SQLite* и д.р.

На официалния сайт на PHP се намира обширна библиотека с информация за езика и модулите му, която може да се използва както за основно запознаване с езика, така и като справочник по време на работата с него.

Поради отворения характер на езика съществуват множество потребителски групи в България и по света посветени на програмирането с PHP, където всеки може да получи помощ в работата си с този език.

PHP Фреймуъркове [14]

В зората на уеб разработката PHP е бил език за програмиране разработен умишлено с улеснен синтаксис с идеята да бъде използван от хора, които не са програмисти за изграждането на прости уеб страници с лична или рекламна цел. С течение на времето, обаче, света на интернет се разширява с огромна скорост, което води до увеличаване на изискванията към уеб разработчиците.

Поради своята опростеност PHP идва със нескромно количество бъгове, недомислици и на моменти не малка тромавост. Със съвременните изисквания, това може да доведе до бавна разработка или бавни сайтове, много безсънни нощи и скъсани нерви.

За щастие обаче в последните години са се разработили множество от тъй наречените PHP Фреймуъркове (*PHP Frameworks*). Фреймуърковете представляват набор от функции, които спестяват часове работа на уеб разработчика. Целта на всеки фреймуърк е да създаде по-лесен и унифициран достъп до части от вградената PHP функционалност и да предостави често използвани функции като готов и лесно достъпен код.

Фреймуърковете също спомагат за писането на структуриран код и създават удобни кодови конвенции, които позволяват на различни програмисти, работещи с един и същи фреймуърк да работят лесно с кода на своите колеги.

Фреймуърковете идват в много цветове (метафорично казано). Използват се различни логики за програмиране при създаването им, но напоследък почти всяка такава софтуерна рамка използва до една или друга степен MVC подхода.

MVC Подход

MVC подхода, всъщност, съществува от доста отдавна и далеч не е създаден специално за уеб разработка. Основната му идея е разделянето на функциите, които отговарят за представянето на информацията пред крайния потребител от тези, които отговарят за манипулирането на споменатата информация. MVC е съкращение от *Model-View-Controller*. Тези три компонента са в основата на една MVC система. Контролерите в една такава система отговарят за това какво и кога ще се изпълни. Чрез тях се извършва извикване на модели зареждане на изгледи и комуникацията между двете. Моделите се използват за връзка с база данни. В тях е хубаво да съществува логика за манипулиране на данните в базата. Накрая всеки изглед (*view*) представлява крайния изход, който потребителя вижда, когато използва нашето приложение.

Ако един фреймуърк използва MVC програмиста ще знае, че винаги може да разчита, например, цялата логика за връзките към базата данни да се намира в папката „models” на системата.

Има голям набор от фреймуъркове, но най-добър фреймуърк няма. Има платени, безплатни, по-леки с по-малко функционалност, по-тежки с повече функционалност и пр. Тук става въпрос за удобство. Всяка една промяна на фреймуърка изисква изучаване на новия, привикване към неговата структура и конвенции, и обикновено тази перспектива не изглежда много привлекателна когато имаш няколко крайни срока зад ъгъла.

База данни [15]

База данни представлява колекция от логически свързани данни в конкретна предметна област, които са структурирани по определен начин. В първоначалния смисъл на понятието, използван в компютърната индустрия, базата от данни се състои от записи, подредени систематично, така че компютърна програма да може да извлича информация по зададени критерии.

Поддръжката на база от данни се осъществява от т.нар. Система за управление на бази от данни (*СУБД*).

Съществуват няколко вида БД:

* Йерархични бази от данни
* Мрежови бази от данни
* Релационни бази от данни
* Обектно ориентирани бази от данни

**Йерархични бази от данни**

Проектът за първата система за бази от данни е бил започнат с цел управлението на данните за програмата Аполо на НАСА. Данните били организирани йерархично, в съответствие с директорийната организация на данните във файловите системи. С времето били забелязани редица проблеми, които възникват при използването на данни организирани по този начин. Например един от основните проблеми е необходимостта от повтаряне на една и съща информация на различни нива. За това в днешни дни този тип организация почти не се използва.

Йерархичния модел организира информацията като структурира повтарящите се групи в нея. Йерархичната БД се състои от подредено множество последователности или по-точно от множество екземпляри на един и същи тип дърво. Зависимите една от друга поредици от данни се разглеждат като самостоятелни единици. Използването им става чрез търсещ ключ на по-горно ниво.

Самата концепция за БД възниква през 60-те години с реализацията на IMS, продукт на IBM, осигуряващ управлението на данни организирани в йерархии.

**Мрежови бази от данни**

Мрежовият модел на базите от данни е измислен от Чарлс Бейчман. През 1973 г. той получава награда на Тюринг за него.

Мрежовият модел е възникнал като мярка за справяне с проблемите на йерархичния модел. Той предоставя възможност за установяване на връзки от тип 1-N между различните нива на йерархията. Връзките могат да съществуват без никакви ограничения. За да се открият определени данни в база организирана по този начин е необходимо да се познава целия път на достъп до тях. Поради тази причина информационните системи използващи мрежови бази са зависими от структурата на данните в тях.

**Релационни бази от данни**

През 1970 г., когато системите базирани на йерархичния и мрежовия модел са били в разгара на развитието си, Едгард Франк Код публикува статия, в която предлага разнородните данни да се съхраняват в таблици, което ще позволи да се установят връзки между тях. В днешно време този модел е масово разпространен, но през 1970 г. тази идея е смятана за интелектуален куриоз. Смятало се, че тези таблици не биха могли да бъдат ефективно управлявани от компютър. Този скептицизъм не успял да спре проучванията на Е. Ф. Код. Един от първите прототипи на система за управление на релационни бази от данни (СУРБД) е бил създаден в лабораториите на IBM.

От 80-те години насам, тази идея се развила и била приета в индустрията. През 1987 г. езикът за заявки към БД, SQL, е стандартизиран. В днешно време, релационният модел е най-масово използвания в системите за управление на бази от данни.

**Обектно ориентирани бази от данни**

Обектно – ориентираните бази от данни (OODB) съхраняват обекти. Тези обекти могат да бъдат лесно намирани и споделени.

OODB осигуряват възможност за съхраняване на сложни типове от данни като изображения, аудио, видео, текст и др., без да е необходимо те да се трансформират в сложни структури от данни за да бъдат съхранени в БД.

Стремежът при OODB е да се разширят ограничените типове от данни, характерни за традиционните модели на бази от данни, със специфични типове от данни.

**Система за управление на бази от данни (*СУБД*)**

Система за управление на бази от данни, СУБД (*Database management system, DBMS*) е набор от компютърни програми, контролиращи изграждането, поддръжката и използването на бази от данни. Примери за такива системи са *Microsoft SQL Server*, *MySQL*, *Access*, *Oracle*, *Paradox*, *dBase*, *FoxPro*, *Cliper*, *Sybase*, *Informix*. Някои от тях са потребителски ориентирани програмни среди, а други са по-скоро езици за създаване и описване на бази от данни. Програмата *Microsoft Access* е типичен пример на потребителски ориентирана програма за управление на бази от данни. Макар че в Microsoft Access е възможно и програмиране с езика SQL, малки бази от данни могат да бъдат създавани и без програмиране.

Системите за управление на базите от данни създават, обработват и поддържат точно определени структури от данни. Съществуват три типа организация на данните и връзките между тях (*наричани модели на бази от данни*): йерархичен (*Hierarchical model*), мрежови (N*etwork model*) и релационен (*Relational model*), като така се различават йерархични бази от данни, мрежови бази от данни и релационни бази от данни. Най-популярен е релационният модел, при който данните се организират в таблици, между които се осъществяват връзки/съотношения (*т.н. релации*). Информатиката познава още 2 вида модела на бази от данни, различни от основните 3, а именно многомерен/многопространствен (*Multidimensional model*) и обектен (*object model*).

Таблиците се състоят от редове и колони. Понякога редовете се наричат записи, а колоните - полета.

MySQL

MySQL е популярна за интернет приложения като *MediaWiki* или *Drupal* и е база данни в *LAMP*, *MAMP* и *WAMP* платформи (*Linux/Mac/Windows-Apache-MySQL-PHP/Perl/Python*), и за софтуера с отворен код *Bugzilla* - приложение за проследяване на грешки. Нейната популярност като Интернет приложение е тясно свързана с популярността на PHP, като често, комбинирани с MySQL, са наречени Динамичното дуо. Лесно е да се намерят много източници, които ги комбинират в интернет статии или книги (*PHP and MySQL for Dummies, PHP and MySQL Bible, Beginning PHP and MySQL, и други*) В тези книги се твърди, че MySQL е по-лесен за изучаване от други бази данни. В пример от „Книгата за глупаци“ се посочва, че от MySQL може да се излезе с командите *exit* или *quit*, но това е вярно и за много други бази данни.

**Администриране**

За администриране на MySQL може да се използва включеното приложение работещо от Командна линия (*Командите: mysql и mysqladmin*). Също достъпни за сваляне от интернет сайта на MySQL са GUI приложения за администриране*: MySQL Administrator и MySQL Query Browser*.

Широкоразпространена и популярна алтернатива, написана на PHP, е софтуерът с отворен код *phpMyAdmin*, *Database* *Master* - *MySQL* *and* *SQLtite* *Manager*

**Платформа**

MySQL работи на много различни Платформи—включващи *AIX, BSDi, FreeBSD, HP-UX, GNU/Linux, Mac OS X, NetBSD, Novell NetWare, OpenBSD, OS/2 Warp, QNX, SGI IRIX, Solaris, SunOS, SCO OpenServer, SCO UnixWare, Tru64, Windows 95, Windows 98, Windows ME, Windows NT, Windows 2000, Windows XP* и други версии на *Windows*. MySQL е прехвърлена и на *OpenVMS*

**Последна версия на продукта**

От началото на 2009 г. MySQL предлага официална версия 5.1.32. Тя има следните възможности:

* ANSI SQL 99 - частично, както и разширения
* Междуплатформена поддръжка
* Съхранени процедури
* Тригери (*Тriggers*)
* Указатели (*Cursors*)
* Обновяеми изгледи (*Views*)
* Пълна поддръжка на VARCHAR
* INFORMATION\_SCHEMA
* Стриктен режим (*Strict mode*)
* Поддържа X/Open XA разпределена обработка на транзакции (*DTP*); запазване в две стъпки (T*wo Phase Commit*) като част от това използвайки InnoDB на Oracle
* Независими хранилища (*Storage Engines*)
* Транзакции с InnoDB, BDB и Cluster хранилища; възстановителни точки (*Savepoints*) използвайки InnoDB
* Поддръжка на SSL
* Кеширане на запитвания (*Query caching*)
* Вложени заявки (*Sub Selects*)
* Репликация от тип "Главен" - "Подчинен" (*master/slave*), много "подчинени" и един "главен", не поддържа автоматично много "главни" за един "подчинен" (*multiple masters per slave*).
* Пълнотекстово индексиране и търсене с MyISAM

# 

# Глава 3

Описание на реализацията на системата

Видове акаунти

Сайтът се състои от две основни части - Клиентска и Административна част. Първата се предоставя за ползване от потребителите, търсещи и/или публикуващи обяви, а втората администраторите и модераторите на сайта, отговарящи за поддръжката на целостта на системата и модериране на вече създадените обяви.

**Потребители от тип Users**

Това са нормалните потребители. Всеки гост на сайтът след като се регистрира получава акаунт от тип Users. След създаването на акаунт и успешното му активиране потребителите ще разполагат с потребителско меню.

Типът Users ще бъде разделен на 2 вида:

* **Employers** (Работодатели)
* **Employee** (Служители)

Всеки акаунт ще има възможността да премине от „Работодателски“ в „Служителски“ и обратното по всяко време.

Потребителите с работодателски акаунт ще имат следните възможности:

* Да публикуват обяви за работа.
* Ще им бъде предоставена страница с всички обяви, които са публикували.
* Ще имат възможността да редактират своите обяви.
* Ще имат възможността да изтриват своите обяви.
* Ще имат списък с всички кандидатури, които подадени за дадена обява.
* Ще имат възможността да одобряват и отхвърлят кандидатури.
* Ще имат възможността да закупят премиум обяви, които ще имат приоритет на показване пред нормалните обяви.
* Потребителите с работодателски акаунт няма да имат възможността да кандидатстват за обяви.

Потребителите със служителски акаунт ще имат възможността да кандидатстват за работа.

**Потребители от тип Moderators (Модератори)**

Модераторите ще бъдат хората, които ще държат нещата под контрол. Ще имат възможността и задачата да контролират обявите на потребителите, да наблюдават за спам и др.

Освен всички функции на обикновените потребители, ще имат възможността да изтриват и редактират всяка обява която не отговаря на общите условия.

**Потребители от тип Administrator (Администратор)**

Това са нашите Администратори. Те са собствениците на системата. Освен всички права които имат обикновените потребители и модераторите, администраторите на сайтът ще разполагат с администраторски контролен панел с помощта на който ще могат да променят външният вид на системата. Ще имат възможността да добавят реклами по сайтът, да променят менютата и ще им бъде предоставен достъп до базата данни и сорс кодът (Source Code) на сайтът.

**Гости**

Гостите ще бъдат нерегистрираните потребители. Гостите ще бъдат най-ограничени откъм права и ще могат единствено да използват търсачката за работа и да разглеждат обявите за работа.

Функционални възможности на системата

Системата ще има следните възможности:

* Регистрация на потребителски акаунти
* Добавяне на обяви
* Редактиране на обяви
* Изтриване на обяви
* Търсене по множество критерии
* Изпращане на кандидатури
* Преразглеждане на кандидатури
* Изтриване на кандидатури

**Администраторски панел:**

* Списък с всички обяви
* Изтриване на обяви
* Редактиране на обяви
* Добавяне на рекламни банери
* Премахване на рекламни банери

Реализация

База данни на системата

Базата данни на системата се състои от 8 таблици.

* Categories (Категории) **–** В тази таблица са запазени всички категории за работа.
* Content (Обяви) – В тази таблица са запазени всички обяви, които потребителите ще създават.
* From (От) – Тази таблица садържа от кого се пуска обявата. (Фирми, Агенции или частни лица)
* Hours (Работно време)
* Kind (Вид) – Вид на работата (Постоянна, временна и пр.)
* Level (Ниво) – За кого е предназначена работата.
* Location (Градове) – Тук се записват градовете
* Users (Потребители) – Тук се записват потребителите
* User\_type (Видове Акаунти) – Таблицата в която се запазват типовете потребители.
* Applications (Кандидатури) – Таблицата в която ще се запишат кандидатурите на потребителите.

**Категории (Categories)**

В тази таблица ще впишем всички категории работи.

*Пример:*

* *Авиация, Летища и Авиолинии*
* *ИТ - Разработка/поддръжка на софтуер*

Таблицата се състои от следните колони:

* Cat\_ID – int(11), AUTO\_INCREMENT, NOT NULL
* Cat\_Name – varchar(100), NOT NULL

**От (From)**

В тази таблица ще впишем от кого се предлага обявата.

*Пример:*

* *Фирми/Организации*
* *Агенции*
* *Частни лица*

Таблицата се състои от следните колони:

* From\_ID – int(11), AUTO\_INCREMENT, NOT NULL
* From\_name – varchar(50), NOT NULL

**Работно време (Hours)**

В тази таблица ще бъде вписано работното време.

*Пример:*

* *Пълно работно време*
* *Непълно работно време*
* *Почасово*

Таблицата се състои от следните колони:

* Hours\_ID – int(11), AUTO\_INCREMENT, NOT NULL
* Hours\_name – varchar(50), NOT NULL

**Вид (Kind)**

В тази таблица ще впишем заетостта на работата.

*Пример:*

* *Постоянна работа*
* *Временна/сезонна*
* *Дистанционна работа*
* *Стажове*

Таблицата се състои от следните колони:

* Kind\_ID – int(11), AUTO\_INCREMENT, NOT NULL
* Kind\_Name – varchar(30), NOT NULL

**Ниво (Level)**

В тази таблица ще впишем експертността.

*Пример:*

* *Мениджмънт*
* *Супервайзъри/Тийм лидери*
* *Служители/Работници*

Таблицата се състои от следните колони:

* Level\_ID – int(11), AUTO\_INCREMENT, NOT NULL
* Level\_Name – varchar(100), NOT NULL

**Град (Location)**

В тази таблица ще впишем всички градове.

*Пример:*

* *Пловдив*
* *София*
* *Хасково*

Таблицата се състои от следните колони:

* city\_ID – int(11), AUTO\_INCREMENT, NOT NULL
* city\_Name – varchar(30), NOT NULL

**Потребители (Users)**

В тази таблица ще се попълват нашите потребители.

Таблицата се състои от следните колони:

* User\_ID – int(11), AUTO\_INCREMENT, NOT NULL
* First\_name – varchar(20), NOT NULL
* Last\_name – varchar(20), NOT NULL
* User\_Email – varchar(50), NOT NULL
* GSM – varchar(15), NOT NULL
* Username– varchar(30), NOT NULL
* Password– varchar(255), NOT NULL
* User\_type\_ID – varchar(30), NOT NULL

**Обяви (Content)**

В тази таблица ще се попълват обявите

Таблицата се състои от следните колони:

* ID – int(11), AUTO\_INCREMENT, NOT NULL
* Thread – varchar(60), NOT NULL
* Content – varchar(8000), NOT NULL
* SalaryFrom – int(50), NOT NULL
* Email – varchar(50), NULL
* Telephone – int(15), NULL
* User\_ID – int(50), NOT NULL
* Kind\_ID – int(50), NOT NULL
* Level\_ID – int(50), NOT NULL
* Location\_ID – int(50), NOT NULL
* Categorie\_ID – int(50), NOT NULL
* Hours\_ID – int(50), NOT NULL
* From\_ID – int(50), NOT NULL
* regDate – Datetime, CURRENT\_TIMESTAMP

**Кандидатури (Applications)**

В тази таблица ще се попълват кандидатурите на потребителите.

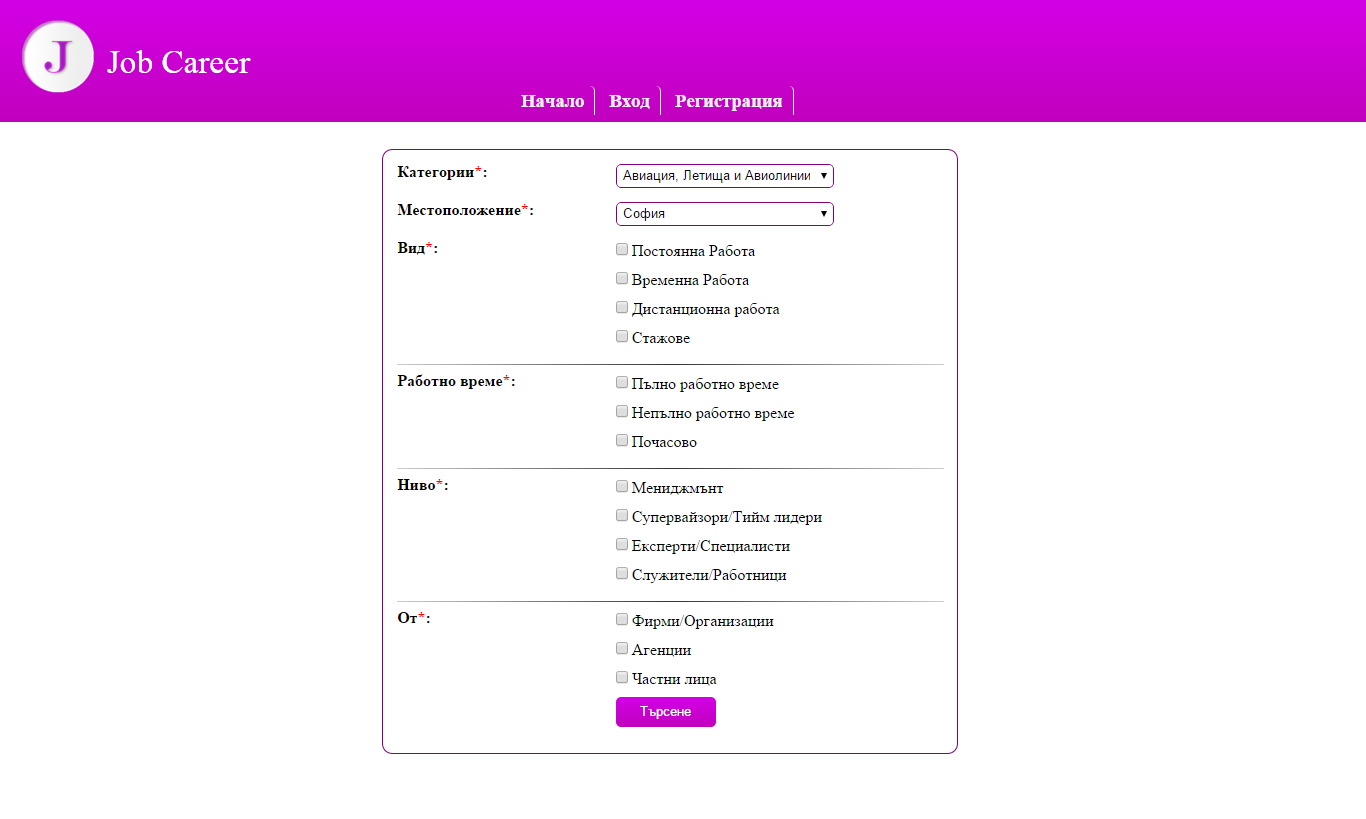
Таблицата се състои от следните колони:

* Application\_ID – int(11), AUTO\_INCREMENT, NOT NULL
* CV\_Link – varchar(50), NOT NULL
* Content\_ID – int(11), NOT NULL
* User\_ID – int(11), NOT NULL
* Note – varchar(2000), NULL
* sendedDate – datetime, Current\_Timestamp

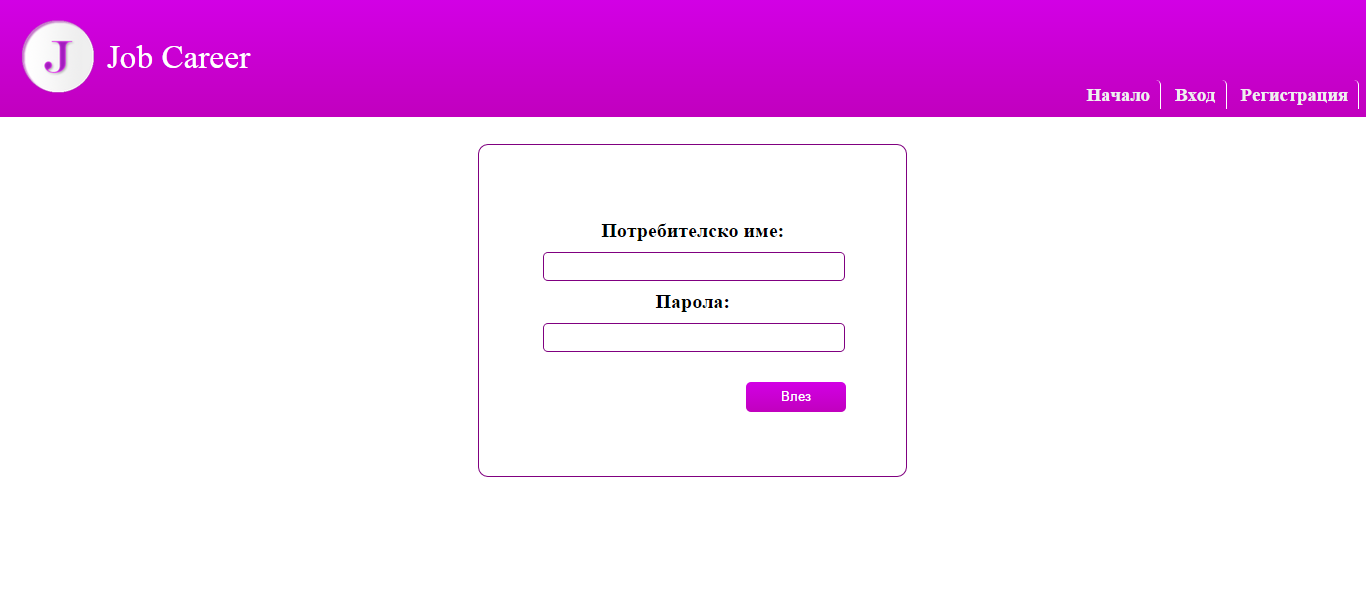
Front-End Реализация

В следващите страници ще се запознаем с потребителският изглед на системата.

**Начална страница**



**Фиг. 1 Начална страница**

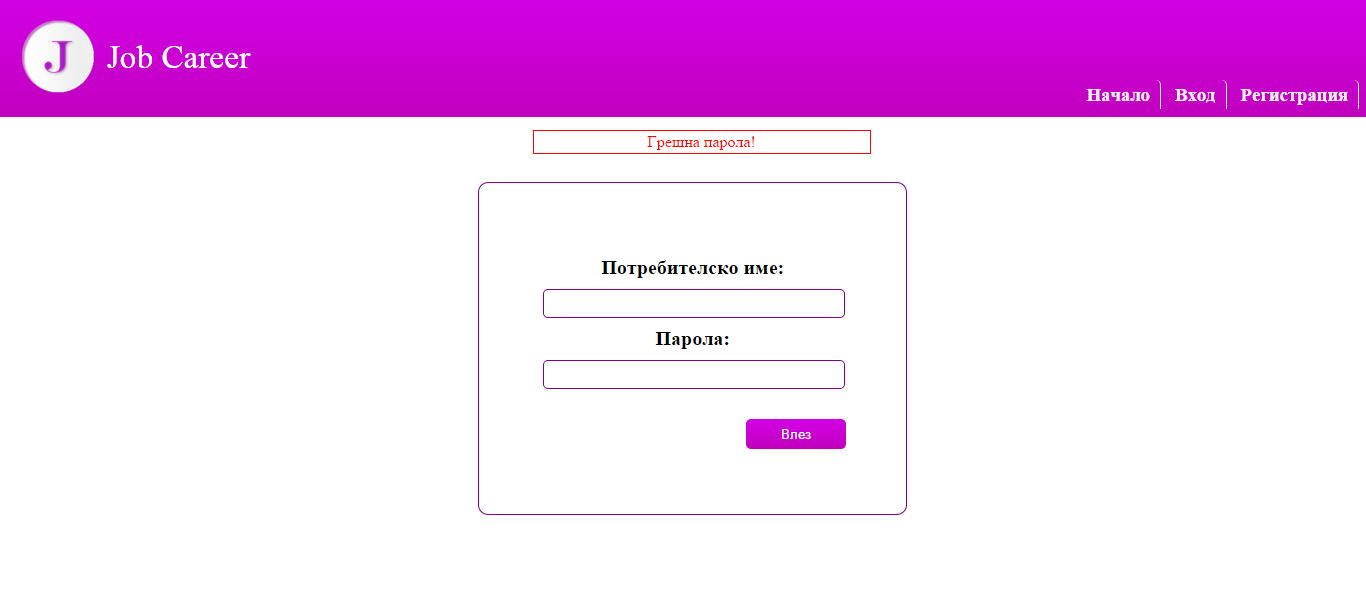


**Фиг.2 Форма за вход**

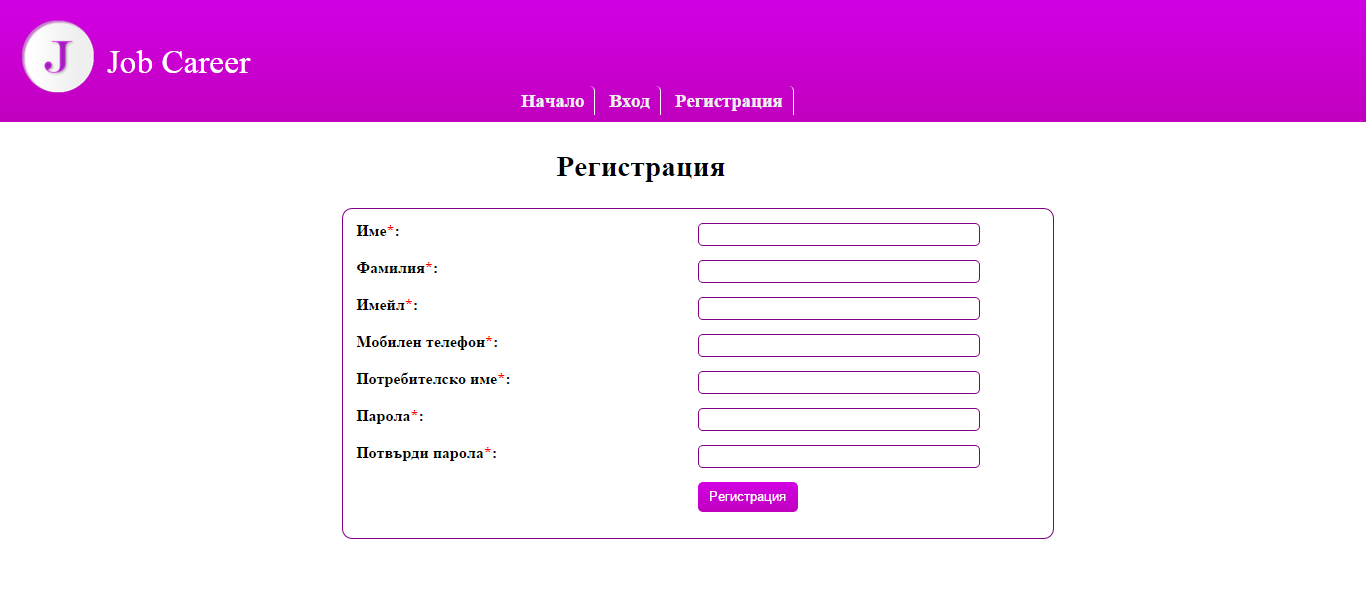
Във формата за вход имаме да въведем 2 неща:

* Потребителско име
* Парола

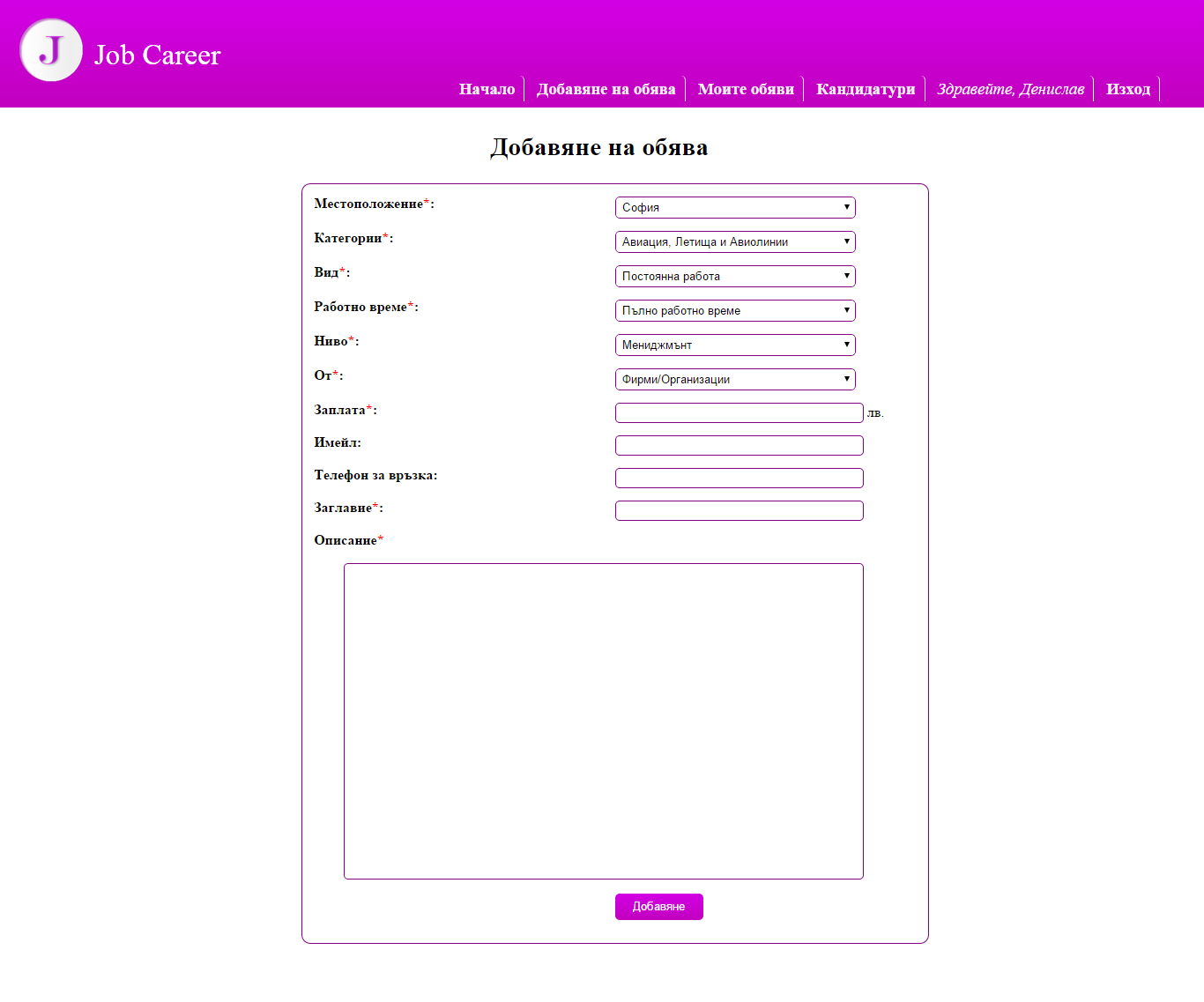
Ако едно от нещата не е вярно, ще ни изкара следната грешка:



**Фиг. 2.1 Грешка за грешна парола**

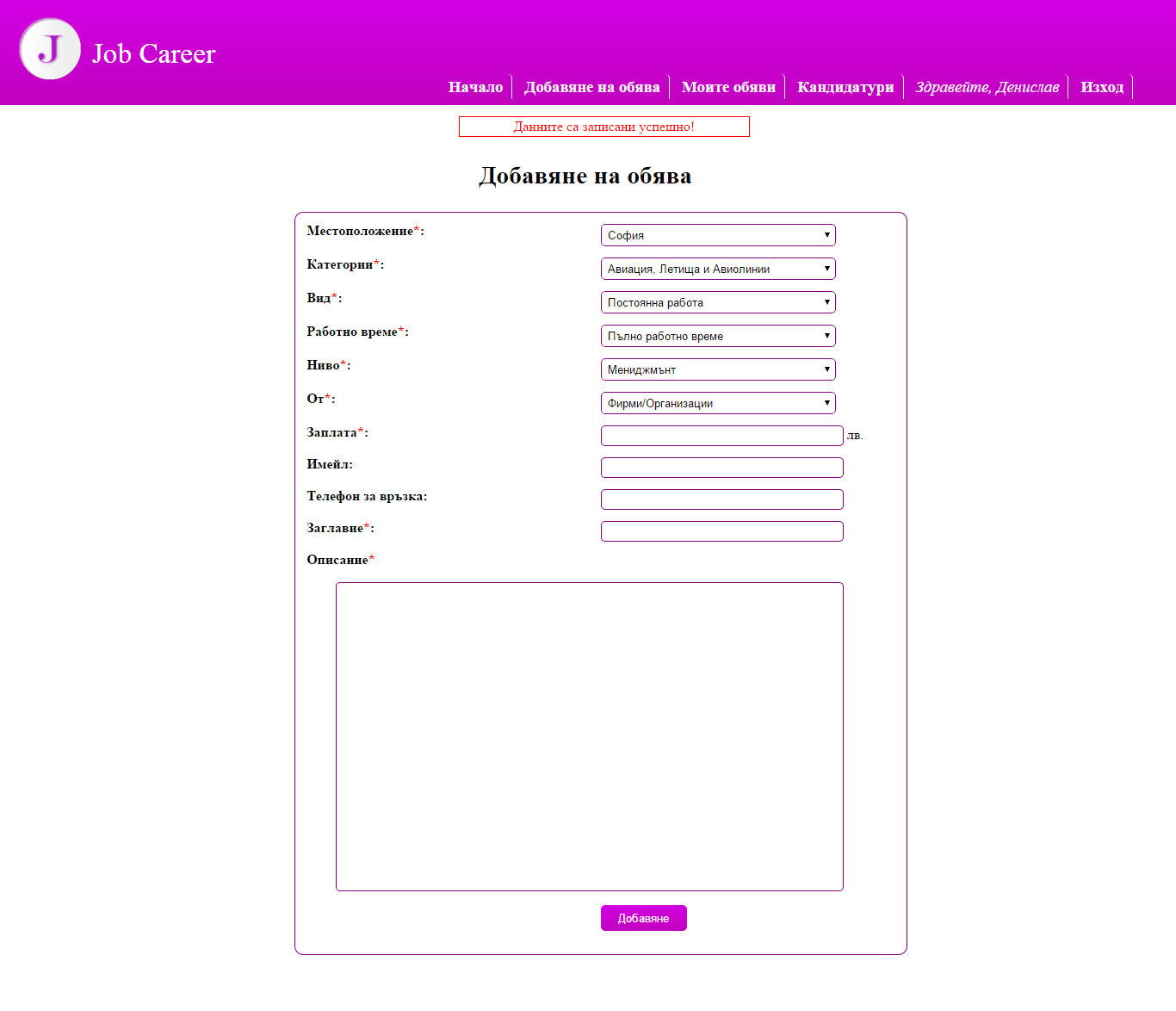


**Фиг. 3 Регистрационна форма**

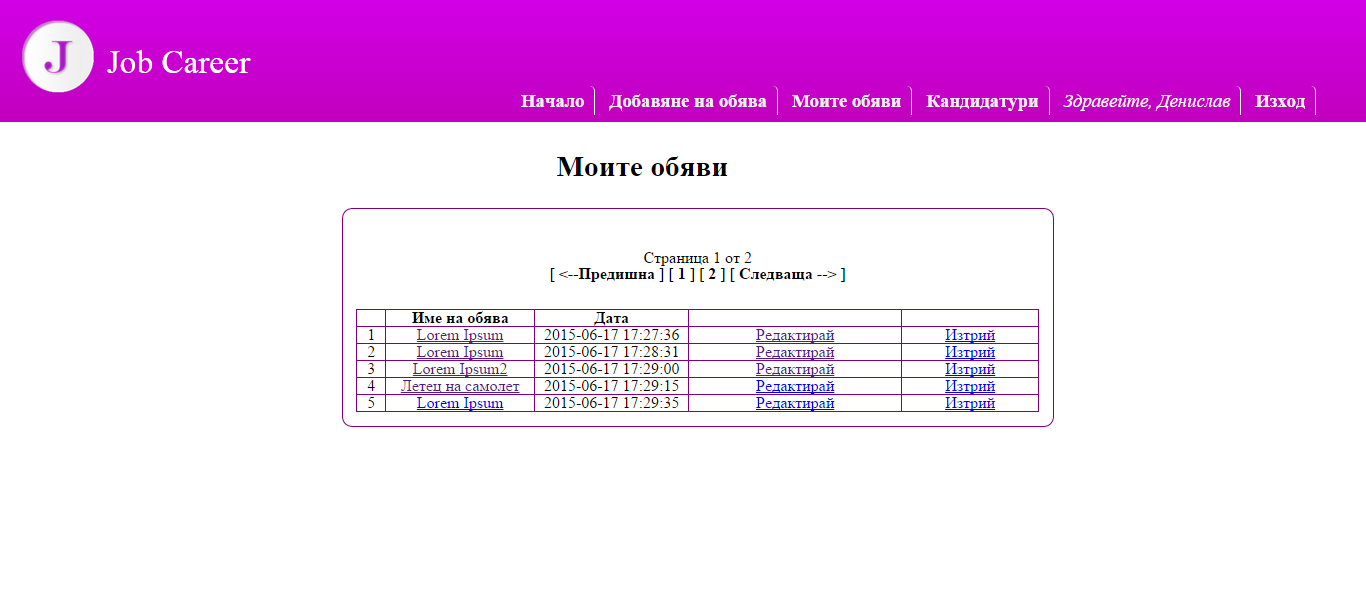


**Фиг. 4 Добавяне на обява**

След като сме влезли в потребителският ни акаунт ще имаме възможността да добавим нова обява. След като сме въвели всички задължителни полета ще ни излезе съобщението на фиг. 4.1

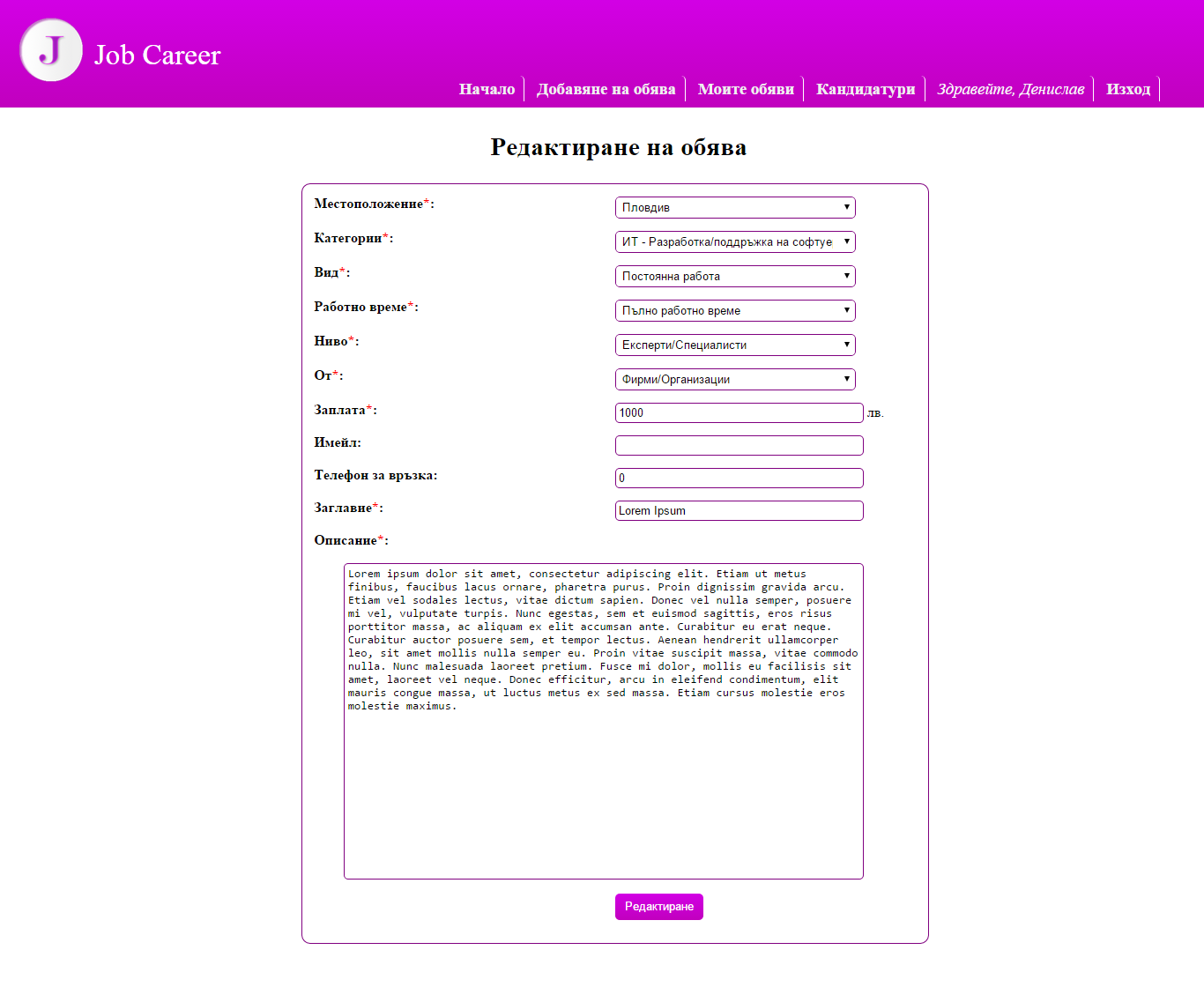


**Фиг. 4.1 Съобщение след добавяне на обява.**



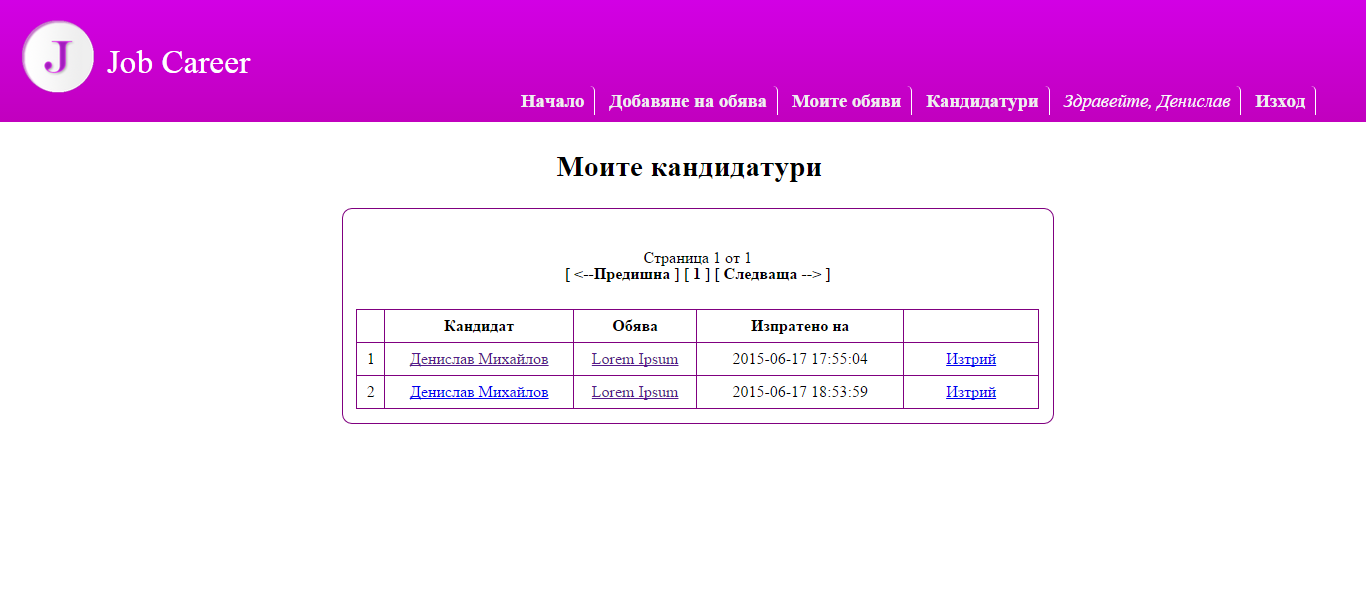
**Фиг. 5 Моите обяви**

Страницата „Моите Обяви“ се състои от една таблица в която са показани всички обяви които сме създали. От тази страница ние можем да редактираме и изтриваме обяви. Също така е добавено и подходящо страниране.



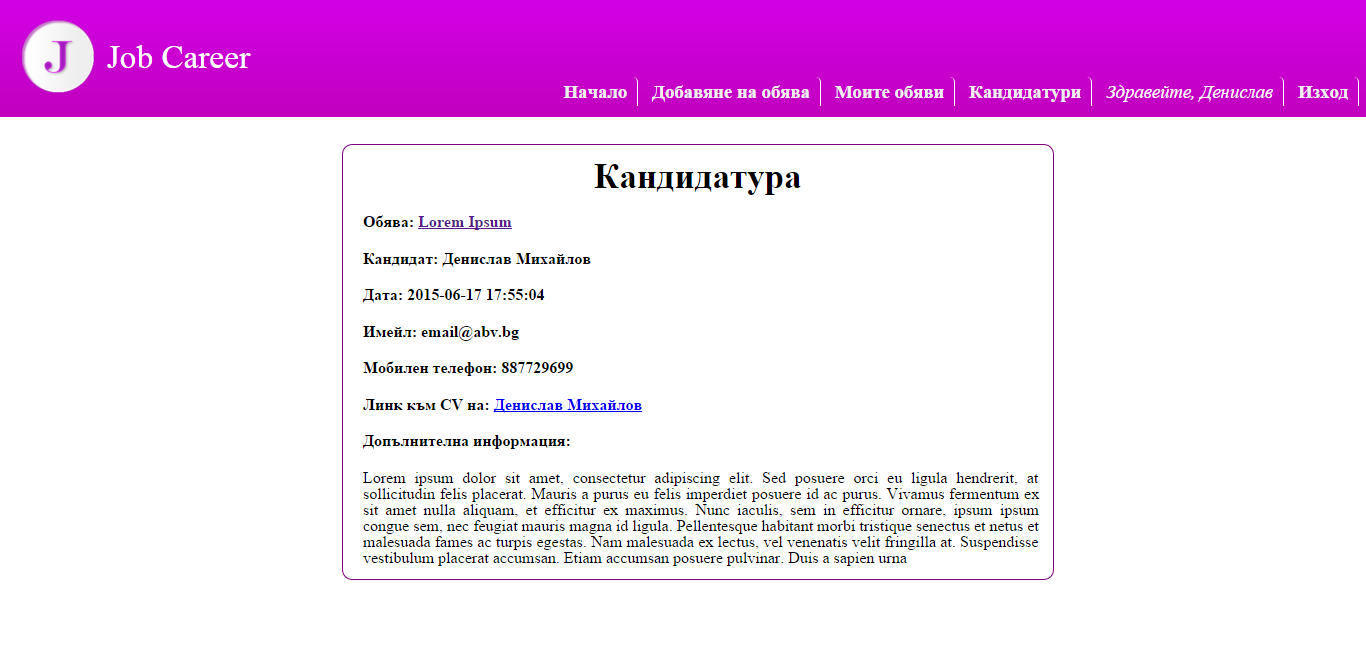
**Фиг. 5.1 Редактиране на обява**

След избиране на желаната обява, която искаме да редактираме, предварително са ни попълнени полетата.

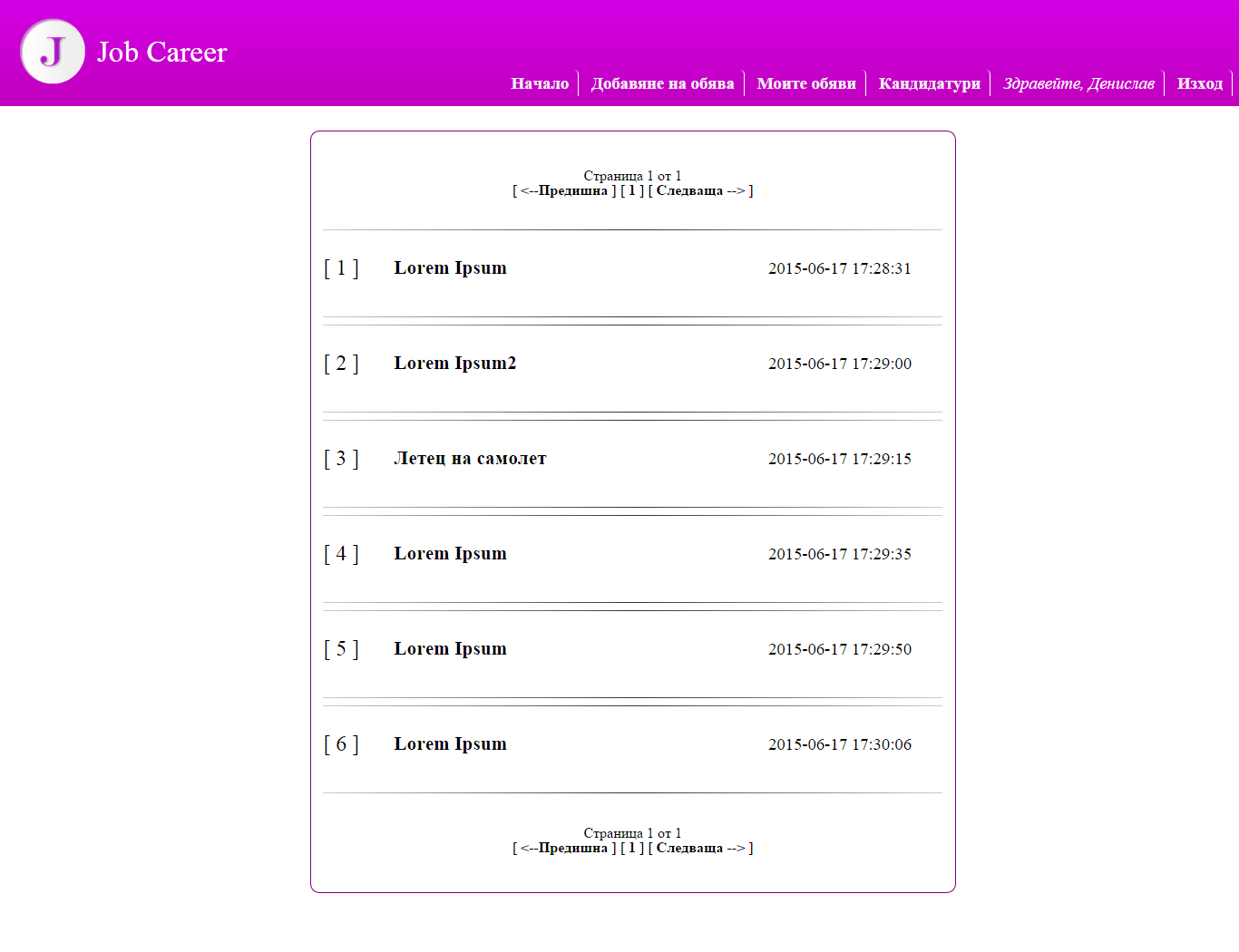


**Фиг. 6 Моите кандидатури**

В тази страница можем да видим всички хора, които са кандидатствали за нашите обяви. След натискането на името на кандидатът ни излиза страница за разглеждане на самата кандидатура. *(фиг. 6.1)*



**Фиг. 6.1 Изглед на кандидатурата**



**Фиг. 7 Резултат от търсенето**

След като натиснем бутонът за търсене на началната страница ще ни изкара списък отговарящите обяви, започващ от най-новите към най-старите. Резултатите бъдат показани по 10 на страница.

След избирането на дадена обяви ще ни се покаже подробна информация за обявата. (*Фиг.7.1*)



**Фиг. 7.1 Изглед на обява**



**Фиг. 7.2 Администраторски изглед на обява**

На фиг. 7.2 се виждат два нови бутона, които ги няма на фиг. 7.1. Добавена е функция за редакция и изтриване на обява, това е функция която само Модераторите и Администраторите притежават, независимо дали те са създали обявата или не.

Сорс код (*Source Code*) на системата

Дотук с потребителският изглед на системата за търсене и предлагане на работа. В следващите страници ще се запознаем с по-интересните работи от сорс кодът.

**Функция за защита от SQL инжекции.**

function clean($str) {

$str = @trim($str);

if(get\_magic\_quotes\_gpc()) {

$str = stripslashes($str);

}

return mysql\_real\_escape\_string($str);

}

**Добавяне на нова обява**

require('conn.php');

include('globalFunctions.php');

if (!isset($\_SESSION['logged'])) {

header("Location:index.php");

}

if (isset($\_POST['add'])) {

if (

empty($\_POST['location']) ||

empty($\_POST['categories']) ||

empty($\_POST['kind']) ||

empty($\_POST['hours']) ||

empty($\_POST['level']) ||

empty($\_POST['from']) ||

empty($\_POST['thread']) ||

empty($\_POST['content']) ||

empty($\_POST['salary'])

){

$error\_message = "Не сте въвели някое от задължителните полета!";

}else{

//Getting entries from Form

$location = clean($\_POST['location']);

$categories = clean($\_POST['categories']);

$kind = clean($\_POST['kind']);

$hours = clean($\_POST['hours']);

$level = clean($\_POST['level']);

$from = clean($\_POST['from']);

$thread = clean($\_POST['thread']);

$content = clean($\_POST['content']);

$user\_id = clean($\_SESSION['user\_ID']);

$salary = number($\_POST['salary']);

$email = clean($\_POST['email']);

$telephone = number($\_POST['telephone']);

// Inserting in to DB

$query = "INSERT INTO content (

Thread,

Content,

SalaryFrom,

email,

telephone,

user\_id,

kind\_id,

level\_id,

location\_id,

categorie\_id,

hours\_id,

from\_id

) VALUES (

'$thread',

'$content',

'$salary',

'$email',

'$telephone',

'$user\_id',

'$kind',

'$level',

'$location',

'$categories',

'$hours',

'$from'

)";

$result = $conn->query($query);

$error\_message = 'Данните са записани успешно!';

}

}

**Моите обяви**

<?php

require('conn.php');

include('globalFunctions.php');

if (!isset($\_SESSION['logged'])) {

header("Location:index.php");

}

// PAGINATION

if (isset($\_GET['page']) && $\_GET['page'] > 0) {

$page = number($\_GET['page']);

}else{

$page = 1;

}

$limit = 5;

$offset = $limit\*($page-1);

$user\_id = clean($\_SESSION['user\_ID']);

$user\_type = clean($\_SESSION['user\_type']);

if ($user\_type > 1) {

$query = "SELECT Thread,ID, regDate FROM content";

}else{

$query = "SELECT Thread,ID, regDate FROM content where user\_id = '$user\_id'";

}

$final\_query = $query." LIMIT $offset,$limit";

$result = $conn -> query($final\_query);

$result2 = $conn -> query($query) or die(mysqli\_errno($conn));

$Qnum\_rows = $result2->num\_rows;

$num\_rows = $result->num\_rows;

$pages = ceil($Qnum\_rows/$limit);

if ($num\_rows > 0) {

while($articles = $result->fetch\_assoc()) {

$Thread[] = $articles['Thread'];

$ID[] = $articles['ID'];

$regDate[] = $articles['regDate'];

}

}else{

$error\_message = "Няма намерени резултати!";

}

**Влизане в системата**

<?php

require('conn.php');

include('globalFunctions.php');

if (isset($\_POST['login'])) {

//Getting information from Form

$username = clean($\_POST['username']);

$password = clean($\_POST['password']);

if (isset($username) && isset($password)) {

$query = "SELECT username, password, first\_name,users\_ID, user\_type\_id FROM users WHERE username = '$username'";

$result = $conn -> query($query);

$num\_rows = $result->num\_rows;

if ($result->num\_rows > 0) {

while($users = $result->fetch\_assoc()) {

$username = $users['username'];

$hash = $users['password'];

$fName = $users['first\_name'];

$user\_ID = $users['users\_ID'];

$user\_type = $users['user\_type\_id'];

}

} else {

$error\_message = 'Грешен потребител!';

}

if ($num\_rows == 1) {

if (password\_verify($password, $hash)) {

$\_SESSION['logged'] = 1;

$\_SESSION['fname'] = $fName;

$\_SESSION['user\_ID'] = $user\_ID;

$\_SESSION['user\_type'] = $user\_type;

header('Location: index.php');

}else{

$error\_message = 'Грешнa парола!';

}

}

}

}

# Заключение

Предимствата от изграждането на кариерна система са много. Едно от най-важните е възможността на работодателите да разглеждат кандидатурите на всеки един кандидат, който би искал да работи в дадената сфера. Друг плюс е това, че имат 24/7 достъп до своите обяви и кандидатури, и могат да бъдат променяни колкото пъти пожелаят независимо дали е в работно време, или не. След създаването на основните функции на една кариерна система е време да помислим и за други функционалности с които бихме могли да я разширим и да улесним нашите потребители

При следващо подобрение на кариерната система, могат да се добавят следните функционалности:

* Рекламни банери
* Администраторски панел за по-лесно поддържане на системата
* Закупване на пакети за промотиране на своите обяви. След като се закупи обявите ще се превърнат в премиум за определен брой от дни и ще се показват на по-челна позиция от нормалните обяви.
* Добавяне на текстов редактор.
* Добавяне на форма за лесно създаване на CV.

# Използвана литература и източници

**Книга:** *Програмиране и Web дизайн с PHP5, MySQL, Apache – авт. Елизабет Нарамор и Джейсън Гернър*

**Книга:** *HTML 5 & CSS 3 - практическо програмиране за начинаещи – авт. Денис Колисниченко*

**http://php.net –** *Официален сайт и справочник на PHP*

**http://w3schools.com –** *Онлайн уеб туториали*

**http://stackoverflow.com/**

[1] **http://Newbusiness.bg –** *Статия на Люба Спасова за The Network*

[2] **http://Jobs.bg –** *№ 1 Кариерна медия за България*

[3] **http://** **Buljobs.bg –** *Специализирана търсачка за работа.*

[4] **http://** **Jobtiger.bg -** *HR компания и кариерен сайт*

[5-7, 9-13, 15] **http://Wikipedia.org –** *Свободна енциклопедия*

[8] **http://zizuweb.com -** *Изработка на уеб сайтове и уеб дизайн с високо качество.*[14] **http://dildesign-studio.com -** *Изработка и поддръжка на уеб сайтове, SEO оптимизация, уеб и респонсив дизайн.*