

Матвеев Евгений

# Как выбрать Систему Дистанционного Обучения



Эта книга создана для руководителей предприятий, директоров по персоналу, руководителей учебных центров и других сотрудников, которые принимают решение по выбору системы дистанционного обучения для своей организации. Также книга может быть полезна представителям учебных заведений, но в первую очередь, она ориентирована больше на корпоративный сектор, чем на академический.

## Оглавление

Введение .....	4
Цена .....	5
Количество внедрений.....	7
Надежность.....	9
Наличие сообщества .....	10
Техническая поддержка.....	11
Дополнительные услуги.....	13
Открытость .....	14
Масштабируемость .....	15
Универсальность .....	16
Функциональность (Расширяемость) .....	17
Персонализуемость.....	18
Распределенное управление.....	19
Заключение.....	20

## Введение

Когда впервые организация задумывается о выборе системы дистанционного обучения, вопросов возникает больше, чем ответов. А главное, не всегда понятно даже с какой стороны подойти. Казалось бы, надо посмотреть, что и как работает у других, и выбрать себе аналогичную систему. Однако ситуация у вас может быть другая, да и внедрения у всех разные оказываются.

Иногда, компания выбирает путь создания собственной системы, но быстро от него отказывается (а иногда медленно, но все равно отказывается). Сейчас на рынке очень много готовых решений, использование которых несравнимо выгоднее создания собственной системы.

Идеальных и полностью универсальных систем не бывает. Чтобы выбрать систему под себя, нужно сначала четко сформулировать требования, необходимые вашей компании. А затем уже под эти требования подобрать более подходящую систему дистанционного обучения.

Каждая организация уникальна. Но есть некоторые универсальные требования, на которые стоит обращать внимание при выборе программного обеспечения. Конечно, одни требования приоритетней для одной компании, другие – для другой. Так или иначе, в этом руководстве собраны основные требования, на которые нужно обращать внимание при выборе системы дистанционного обучения.



## Цена

Очень часто первое, на что обращают внимание заказчики при выборе системы дистанционного обучения, это цена. Именно поэтому и в нашем списке требований этот пункт стоит на первом месте. Однако, как бы странно это не звучало, на цену нужно обращать внимание в последнюю очередь.

Нельзя цену делать определяющим критерием. Как говорится, скупой платит дважды. В предельном случае система вообще может быть бесплатной. Существует специальное программное обеспечение, которое официально бесплатное. Систем дистанционного обучения, распространяемых таким же путем тоже не мало. Однако бесплатность самого программного обеспечения еще не гарантирует нулевых затрат при его внедрении. Существуют случаи, когда внедрение бесплатных систем обходилось для заказчиков на порядки дороже, чем использование коммерческих систем.

Кроме того, попытки сравнить разные системы от разных разработчиков по стоимости, как правило, безуспешны. Дело в том, что системы могут быть очень разными и по функционалу, и по техническим особенностям, и по возможностям расширения и собственной доработки, и по удобству и гибкости управления, и по особенностям лицензирования, и по количеству и качеству сопутствующих услуг, и по массе других критериев. Вот и цена у всех разная. Чтобы корректно сравнить разные системы по цене, их сначала придется привести к общему знаменателю по всем другим критериям – а это уже совсем не представляется возможным.

Так что, цена – дело десятое. Нужно оценивать систему по другим критериям, которые будут упомянуты ниже, а на цену смотреть в самую последнюю очередь – когда уже выбираете из финалистов при прочих равных. Но до этого, как правило, не доходит – с учетом остальных критериев финалист остается один.

Но как же быть, если у организации урезанные бюджеты, и нет финансов на дорогое коммерческое программное обеспечение? На это счет есть несколько соображений:

Во-первых, если у вас нет денег на подходящую для вас систему, возможно, вообще стоит отложить приобретение? Ведь купить программное обеспечение, дело самое простое – стоимость внедрения и обслуживания, как правило, на порядки выше приобретения. Цена программного продукта, по сравнению с



общей стоимостью проекта электронного обучения в компании, оказывается настолько ничтожна, что ей можно пренебречь, как погрешностью. Так что если не хватает даже на само ПО, то стоит пока вообще отказаться от проекта.

Во-вторых, было бы желание, способы всегда можно найти. Если вы выбрали подходящую для себя систему, но по какой-то причине, никак не укладываетесь в бюджеты, всегда можно договориться с поставщиком. В адекватных рамках поставщик всегда пойдет навстречу. Например, может подготовить для вас несколько усеченный вариант системы по более низкой цене, или может, вы договоритесь об аренде системы, или о рассрочке платежа – да мало ли еще вариантов, было бы желание.

Одним словом, не нужно обращать внимание на цену в первую очередь. Иначе существует вероятность, внедрив более дешевую систему, через некоторое время переходить на другую, более подходящую, систему. И тогда никакой экономии не получится – придется заплатить уже за две системы. Плюс сам переход всегда очень болезнен и недешев. Проще и дешевле сразу начать пользоваться «нужной» системой, чем потом исправлять ошибки.



## Количество внедрений

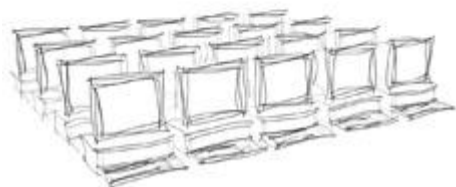
Количество организаций, которые уже пользуются конкретной системой дистанционного обучения, является одним из главных показателей ее функциональности и надежности. Кроме того, чем больше внедрений, тем больше специалистов на рынке, работающих с данной системой. И это позволит положительно решить кадровый вопрос, если не сейчас, то в будущем.

При выборе системы необходимо обязательно узнать, кто находится в числе клиентов у поставщика. Причем, кроме общего количества клиентов, нужно посмотреть, сколько клиентов именно из вашей сферы.

Совершенно недостаточно просто найти на сайте поставщика в рубрику «Клиенты» с перечнем организаций, использующих систему. Не всегда эта информация может быть актуальной. Запросите информацию у поставщика о клиентах из вашей сферы и узнайте, какие особенности внедрения у них были. Запросите описания подробных внедрений. Запросите отзывы и рекомендации клиентов. Договоритесь с какими-либо клиентами из арсенала поставщика лично пообщаться, и подробно обо всем расспросите последних. Как правило, надёжному поставщику скрывать нечего, и он с удовольствием поделится контактами клиентов, и даже организует встречу.

Еще один хитрый косвенный способ проверить популярность системы, это ввести в строке поиска ее название на популярных рабочих сайтах – особенно, если на этих сайтах можно делать поиск и по архивным вакансиям и резюме. Таким образом, можно косвенно судить о популярности той или иной системы – чем больше вакансий с требованиями знаний этой системы и чем больше резюме с описанием соответствующих навыков, тем популярнее система. Кроме того, таким способом можно найти организации, использующие систему, которую вы рассматриваете для себя, и связаться с этими организациями.

Кроме того, существуют другие косвенные способы определения популярности системы. Можно посмотреть, как часто само программное обеспечение или название производителя упоминается в тематических СМИ. На каком месте стоит сайт производителя в Яндексе и Гугле по запросам «дистанционное обучение», «e-learning» и пр. Как часто участвует производитель в тематических выставках и конференциях, что пишут о системе в независимых исследованиях и специализированных журналах.



Сколько вообще лет существует компания-производитель и много ли у нее партнеров. И так далее. По этим и другим косвенным признакам можно судить о популярности системы.

Как показывает практика, чем больше клиентов, тем проще будет работать с программным обеспечением, так как есть с кем посоветоваться, обменяться опытом, решать какие-то задачи совместными усилиями.

Поэтому обязательно посмотрите, насколько популярна система. Конечно, есть вероятность, что некая «молодая» система заткнет за пояс «сторожил», но риски могут оказаться слишком высоки.





## Надежность

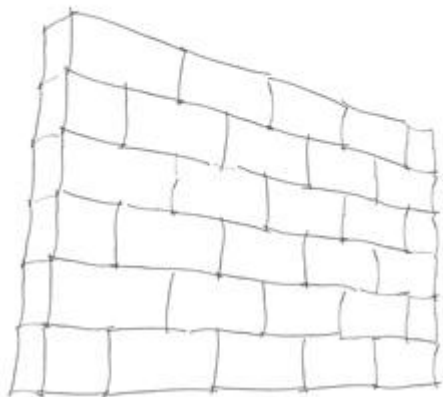
Не бывает абсолютно идеального программного обеспечения, которое работает всегда, у всех и без сбоев. Особенно, если речь идет о сложных крупных программных продуктах. Однако очевидно, что для надежной работы, система должна работать с минимальным количеством сбоев и ошибок. И сбои эти не должны быть настолько критичны, чтобы полностью парализовать работу системы.

К сожалению, бывает, очень сложно оценить надежность программного обеспечения, пока его не внедришь. А это уже чревато немалыми затратами. Поэтому оценивать надежность тоже надо заранее. В полной мере этого сделать, увы, не получится, но кое-что организовать можно.

Во-первых, обязательно установите демо-версию. У любого полноценного программного продукта она должна быть. Демо-версию лучше пытаться использовать в условиях, максимально приближенных к боевым насколько это возможно. Нагружайте демо-версию по полной. Попробуйте весь необходимый функционал, требуемый в реальной работе. Если уже на этом этапе будут сбои и проблемы, то это повод серьезно задуматься о целесообразности приобретения такой системы.

Второй подход – спросить у клиентов, уже использующих систему, которая вас заинтересовала. Причем лучше выбрать клиентов с количеством пользователей, сопоставимым с вашим случаем, и со схожими задачами. Ведь одна и та же система, в которой работают 10 пользователей, может показывать вполне хорошую надежность, а вот при 10 000 пользователей уже будет работать нестабильно. То же самое касается задач, которые решаются с помощью одного программного продукта в разных компаниях. Например, в одном случае система используется только для обучения с помощью электронных курсов, в другом – еще и с онлайн трансляциями в реальном времени. Естественно, в плане надежности и стабильности работы в этих случаях одна и та же система может вести себя тоже по-разному.

Так или иначе, надо использовать все возможности предварительной проверки надежности системы, поскольку это один из самых важных критериев.



## Наличие сообщества

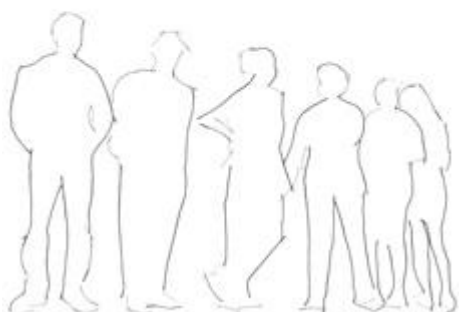
Когда музыкальный коллектив или эстрадный исполнитель становятся популярными, у них появляются сообщества фанатов. Эти фан-группы общаются в интернете на специализированных сайтах, обмениваются информацией о любимом исполнителе, а также встречаются на концертах своих кумиров.

Возможно, для кого-то покажется это удивительным, но у программных продуктов бывают похожие сообщества. Конечно же, это далеко не фан-сообщества поп-исполнителей, но некую аналогию провести можно. Вокруг программных продуктов также могут организовываться сообщества, только они, как правило, узкопрофессиональные и организуются самими производителями программного продукта при достаточном количестве клиентов.

Как же могут выглядеть такие сообщества? В первую очередь, это некий интернет ресурс, на котором все пользователи программного продукта общаются между собой и с представителями разработчика на различных форумах, получают необходимую информацию по программному продукту, знакомятся с документацией и многое другое. Кроме общения в онлайн, у клиентов также есть возможность участвовать в очных встречах, где также можно обмениваться опытом.

Само наличие такого сообщества есть далеко не у всех. Но важно, чтобы оно все-таки было. Во-первых, это косвенно свидетельствует о популярности продукта, во-вторых, при возникновении каких-либо проблем или нестандартных задач, вряд ли вы останетесь в одиночестве, в-третьих, если производитель озадачился созданием и поддержкой такого сообщества, значит, ему нечего скрывать, и велика доля вероятности, что система качественная.

Но только самого факта, что такое сообщество по интересам существует, бывает не достаточно. Если есть возможность хотя бы оценить сайт этого сообщества, то обязательно надо сделать это. Посмотрите насколько активно там идет обсуждение по разным вопросам, сколько всего участников, какие в основном проблемы обсуждаются в форумах и насколько быстро решаются, какие вопросы поднимаются в блогах, есть ли обучающие курсы по системе и в каком они формате, насколько подробная документация представлена, какие еще полезности представлены на сайте сообщества, есть ли очные встречи участников сообщества и так далее. Все эти косвенные вопросы позволяют сформировать более полную картину о системе.



## Техническая поддержка

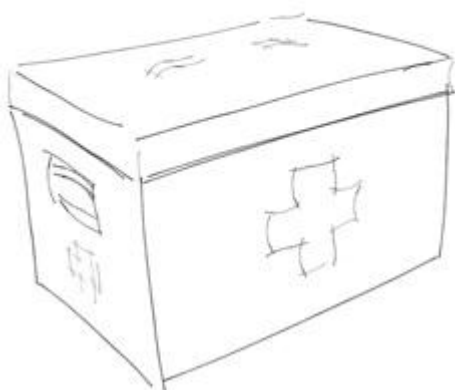
Совершенно очевидно, что при эксплуатации сложного программного обеспечения без технической поддержки вряд ли удастся обойтись. Поэтому важно, чтобы она была и была надежной и оперативной.

Итак, что нужно узнать у поставщика программного обеспечения относительно технической поддержки: Что включает в себя в целом техническая поддержка – консультации по телефону, по электронной почте, удаленное решение проблем, выезды к клиенту? Какое время работы у службы технической поддержки – 24 часа в сутки без выходных или по определенному графику? Каковы сроки реагирования на критические и некритические сбои в системе, а также сроки решения этих проблем? Эти и другие вопросы позволят сложить первоначальное впечатление о технической поддержке.

Также тут немаловажный вопрос стоимости технической поддержки. Как правило, стандартные условия включают в себя бесплатную техническую поддержку в течение года после покупки программы, а начиная со второго года – 20% от стоимости программного обеспечения в год.

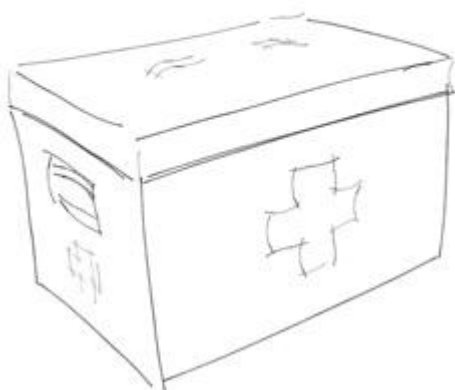
Однако такой, наиболее часто встречающийся способ оплаты услуг по технической поддержке, встречается не всегда. У некоторых поставщиков бывают и другие схемы. Например, бывает, что техническая поддержка стоит не 20, а 30% от стоимости программного обеспечения. Причем, эти 30% надо дополнительно оплачивать с первого года. Причем, платеж тот обязательный. В итоге получается, что мы выбираем более дешевую систему, но когда дело доходит до подписания договора, по факту получается, что цена поднялась на 30% и оказалась выше, чем у «как бы» более дорогой системы.

Другой случай, который тоже может иметь место: Есть одна система с определенной стоимостью, есть другая – у которой функционал аналогичен, но стоимость на порядок дешевле. Наивный покупатель может выбрать второй вариант, так как он представляется наиболее выгодным. Однако на деле все может оказаться совершенно не так. У изначально более дорогой системы в стоимость будет включена техническая поддержка в течение года, а начиная со второго года эксплуатации эти услуги будут составлять не более 20%, причем и они не являются обязательными. В более же дешевой системе за техподдержку



придется заплатить дополнительно сразу, к тому же стоимость ее гораздо дороже, к тому же перечень работ, включенный в тех.поддержку, будет несоизмеримо меньше, к тому же тех.поддержка окажется обязательной услугой, без которой программа откажется работать, к тому же... Одним словом, второй вариант выйдет гораздо дороже первого, просто оплата будет размазана во времени.

Поэтому крайне важно заранее выяснить все нюансы работы технической поддержки, так как это одна из самых важных и, необходимых услуг, которые придется заказывать у поставщика.



## Дополнительные услуги

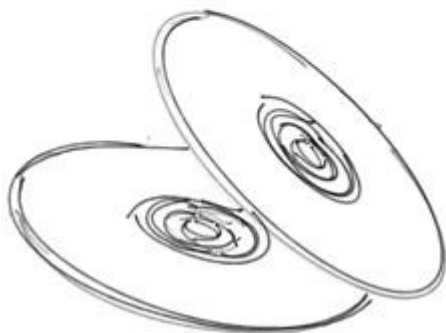
Кроме услуг по технической поддержке, которые являются, как правило, самыми необходимыми, желательно, чтобы у разработчика программного обеспечения или его партнеров были и другие услуги.

Услуги могут быть совершенно разнообразными. Например, расширенные варианты технической поддержки, семинары по работе в системе, разработка электронных курсов, создание дизайнов порталов дистанционного обучения и дизайнов курсов, настройка и доработка системы под задачи заказчика, наполнение портала дистанционного обучения контентом, актуализация электронных курсов, предоставление своих серверов под использование системы и другие.

Услуг может быть много, и они могут быть разнообразными. Вполне вероятно, что эти услуги в данный момент времени, и даже в обозримом будущем, вам вовсе будут не нужны. Однако важно, чтобы они в принципе были, чтобы у поставщика были возможности их оказывать.

Во-первых, это косвенно свидетельствует о зрелости и надежности поставщика, а значит и самого программного продукта. Во-вторых, если дополнительные услуги вам не нужны сейчас, это не значит, что они не понадобятся в будущем. Сегодня все необходимые работы вы можете выполнять внутри компании, но, как знать, завтра ситуация может поменяться, и иметь подстраховку со стороны поставщика никогда не помешает.

Услуги могут быть двух типов. Либо это работы, которые выполняет поставщик на этапе внедрения системы, либо уже в процессе эксплуатации. Первый вариант – это необходимые услуги, которые должны быть в арсенале поставщика в любом случае, даже если вы собираетесь самостоятельно проводить внедрение системы или оно вовсе не нужно. Услуги же, которые могут потребоваться в процессе эксплуатации системы (за исключением технической поддержки), это опциональный набор. У одних поставщиков их может быть много, у других – их вовсе может не быть. Само наличие или отсутствие таких услуг ни о чем не говорит, но по причинам указанным выше, лучше чтобы они все же были, и чтобы их выбор был достаточен.



## Открытость

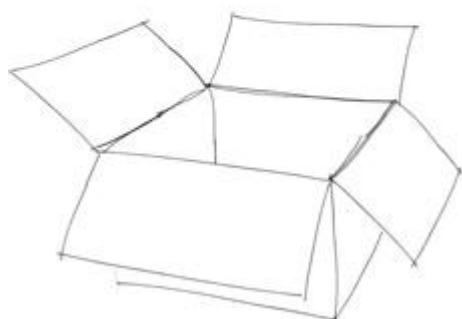
Представим себе ситуацию, когда через какое-то время пользования системой, вам стало не хватать базовой функциональности, и вы решили кое-что переделать. Речь не идет о кардинальном изменении системы, но возможно надо настроить дополнительные шаблоны на сайте, незначительно изменить какой-то функционал или переделать системные сообщения, выдаваемые в тех или иных случаях пользователям. Тогда вы обращаетесь к поставщику, и в случае существования такой возможности (наличия дополнительных услуг, о чем мы говорили ранее), поставщик реализует ваши требования.

Но что если стоимость таких работ не будет укладываться в ваши бюджеты, или у поставщика не будет свободных ресурсов в данный момент, или будут еще какие-либо причины невозможности заказа таких работ этому поставщику? В этом случае должна быть возможность сделать все необходимые доработки либо собственными силами, либо силами других подрядчиков.

Для этого система должна быть, во-первых, максимально гибка в настройках и в управлении, а во-вторых, иметь открытый код и давать возможность самостоятельного его исправления. Это относится и к изменению интерфейса, и к доработкам функционала.

Причем надо иметь в виду, что существует гибкость системы, то есть возможность гибко настраивать систему без доработок кода. А существует именно возможность доработки кода системы, поскольку какие бы гибкие настройки в системе не были, стопроцентной гибкости не бывает.

Поэтому перед приобретением системы нужно обязательно поинтересоваться у поставщика, насколько открыт код системы, что можно безболезненно исправлять, в какой степени можно менять дизайн системы, функционал и интерфейс, какие языки программирования при этом используются. Кроме того, очень важно узнать насколько задокументированы возможности вносить изменения в код системы. Поскольку, если код открытый, но нет никакой документации какие функции и переменные за что отвечают, то править такой код будет крайне сложно. Такая открытость будет лишь условной.



## Масштабируемость

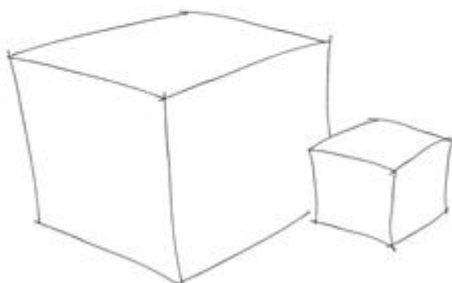
Каких масштабов должна быть система дистанционного обучения? Сколько пользователей она должна поддерживать? Сколько организаций может учитываться в одной системе? Насколько территориально-распределенная работа может быть организована в системе? Сколько сайтов дистанционного обучения может поддерживаться системой?

В идеале на все эти вопросы должен быть один ответ: бесконечно.

Современная система должна быть масштабируема по всем критериям. В ней должно поддерживаться любое количество пользователей без каких-либо доработок системы или изменения ее функциональности. Она должна поддерживать работу неограниченного количества сайтов. Она должна поддерживать территориально распределенную работу.

Причем с территориальной распределенностью не все так просто, как может показаться на первый взгляд. Понятно, что пользоваться системой должны иметь возможность пользователи из любой точки интернета (ну или удаленно из локальной сети). На то она и система дистанционного обучения – удаленное подключение заложено уже в названии. Вот только пользователи, как правило, подключаются к центральному серверу, где и установлена система. Но бывают ситуации, когда связь в слишком отдаленных точках крайне нестабильна и ненадежна или пользователей так много, что один сервер может не справиться. В этом случае система должна поддерживать работу распределенных серверов. То есть одни пользователи автоматически подключаются к одним серверам, другие – к другим. И с определенной периодичностью эти сервера синхронизируются. Пользователям же эта кухня не заметна – визуально они все пользуются одним и тем же порталом дистанционного обучения.

Так что система должна быть масштабируема сразу по ряду критериев. Конечно, в данный момент у вас может не быть такой необходимости и масштабных целей и задач. Но если вдруг в будущем возникнет такая необходимость, система должна это уметь делать без каких-либо глобальных доработок. Кроме того, если система масштабируема, то это также может являться косвенным признаком ее состоятельности.



## Универсальность

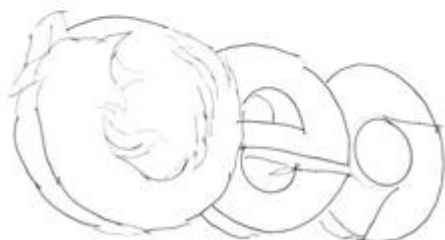
Система должна быть универсальна с технической точки зрения. Конечно, на 100% универсальных программ не бывает, но надо внимательно смотреть, насколько универсальна может быть та или иная система.

Вот что подразумевается тут под универсальностью: кроссбраузерность и кроссплатформенность.

Причем кроссплатформенность, на самом деле, не так сильно важна. Конечно, хорошо, когда система может работать на разных платформах, на разных операционных системах – это несомненный плюс системы. Однако на практике достаточно и одной платформы – той, которая отвечает именно вашим задачам. Надо пояснить – речь идет именно о серверной части программного обеспечения, не видимой пользователям. Что касается операционных систем, которые установлены на компьютерах у обычных пользователей – в этом смысле, конечно, система должны быть кроссплатформенной.

А вот кроссбраузерность должна в идеале поддерживаться максимальная. Поскольку любая система дистанционного обучения, это браузерное решение, то есть с точки зрения пользователя, это обычный интернет-сайт, который работает через обычный браузер, то и работать система должны с максимальным количеством основных браузеров. Причем работать система должна в разных браузерах без установки дополнительных плагинов и надстроек.

В этом смысле универсальность системы очень важна. Однако тут есть и другая сторона медали. Чем с большим количеством платформ и браузеров работает система, тем она сложнее с технической точки зрения. А чем сложнее система, тем больше вероятности сбоев и ошибок. Конечно, не обязательно и не всегда. Но чем универсальнее система, тем пристальнее нужно уделить внимание к ее надежности и стабильности работы.





## Функциональность (Расширяемость)

Так сложилось, что системы дистанционного обучения давно перестали выполнять функцию только дистанционного обучения. Сейчас, как правило, это несколько большее.

В какой-то момент стало очевидным, что кроме функционала непосредственно дистанционного обучения в системе должны быть и другие функции. Например, оценка персонала необходима, когда мы хотим оценить, кого и чему обучать или как повлияло обучение на сотрудника. Таким образом, в системе появился дополнительный программный модуль – оценка персонала.

Затем стало очевидно, что неплохо было бы в одной системе еще и автоматизировать очное обучение – с базой преподавателей, с составлением расписания, с распределением аудиторий и так далее. Тем более, с помощью этого инструмента можно организовывать и смешанное обучение – состоящее и из дистанционной формы, и из очной.

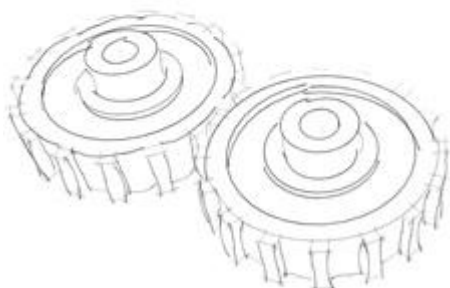
Постепенно система обрастает дополнительным функционалом и становится не просто системой дистанционного обучения, а некой комплексной системой, где дистанционная форма обучения – лишь ее малая часть.

Сегодня большинство современных систем имеет именно обширный функционал. Тут и упомянутые выше оценка персонала, и автоматизация очного обучения. Тут и управление знаниями, и управление кадровым резервом, и подбор персонала, и работа с электронной библиотекой, и электронный деканат для ВУЗов, и автоматизация многих других функций.

Чаще всего, в современных системах все эти функции выделены в отдельные модули. А заказчик может выбрать нужные модули, и как конструктор собрать под себя систему в зависимости от задач.

Чем больше у системы разнообразных модулей, тем лучше. Даже если сегодня вам некоторые модули совершенно не нужны, они могут понадобиться завтра. Поэтому преимущество будет за более функциональной системой.

Поэтому очень важно перед приобретением системы узнать насколько она функциональна (даже если какие-то функции кажутся совсем лишними), как быстро можно расширить эти функции, и как часто и по какому принципу появляются новые модули в системе.



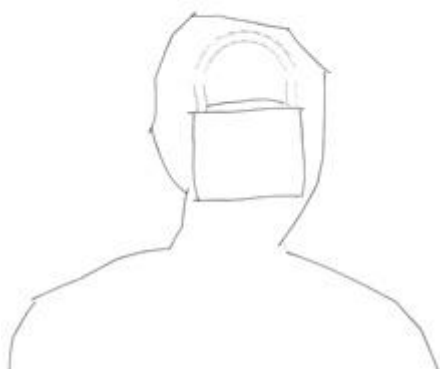
## Персонализуемость

Современная система дистанционного обучения должна уметь гибко подстраиваться под каждого пользователя. Это касается и непосредственных пользователей, обучающихся в системе, и администраторов, которые этой системой управляют.

У вас должна быть возможность очень гибко назначать права доступа для каждого пользователя. В зависимости от того какая должность у пользователя, в какие группы он входит, какой ему назначен уровень доступа и какая роль, а также по ряду других условий, этот пользователь будет видеть только определенные страницы портала, которые он должен видеть, и выполнять там только определенные функции. Например, если вы рядовой пользователь, то вы сможете проходить на портале обучение и тестирование, а также видеть подробную статистику по результатам на специальной странице в личном кабинете. А если же вы руководитель, то кроме этого для вас будет виден и другой раздел, на котором вы сможете просмотреть подробную статистику по своим подчиненным сотрудникам, а может даже и назначить им определенные электронные курсы или тесты. Или же если вы преподаватель, то сможете сделать аналогичные действия с группой прикрепленных к вам обучаемых.

Кроме того, хорошо, если система поддерживает многосайтовость. Одной группе пользователей доступен один сайт, другой группе – другой. Если же сама структура разделов сайтов аналогична для всех групп пользователей, то один и тот же сайт для разных пользователей должен уметь отображаться в совершенно разных дизайнах. Плюс к этому все сайты системы должны поддерживать многоязычность – с возможностью переключения языка пользователями.

Таким образом, система должна уметь настолько гибко работать с пользователями, что можно для каждого отдельного обучаемого организовать свой индивидуальный ресурс с персональной структурой, дизайном, контентом и функционалом. Конечно, на практике такая необходимость вряд ли когда возникнет, но возможности такие должны быть. Делать отдельный персональный ресурс для каждого пользователя, это конечно, непозволительное расточительство, но сформировать отдельные группы пользователей, которым будут доступны разные ресурсы и разный функционал – это вполне реальная задача, которая рано или поздно может возникнуть.



## Распределенное управление

Любая система нуждается в управлении. Пользователи, которые управляют системой, называются администраторами. Хорошая система должна позволять работу не одного администратора, а нескольких. В идеале, таких администраторов должно быть бесконечное количество. Причем возможности у этих администраторов должны быть разными.

Функционал администраторов системы должен быть распределен. Например, один администратор должен иметь возможность только загружать в систему электронные курсы и тесты и назначать их обучаемым. Другой администратор – может только управлять структурой учебного портала, формируя его структуру, наполняя контентом и разграничивая права доступа к разным разделам. А третий – выполняет чисто технические функции, имея только доступ к настройкам системы.

Кроме того, даже администраторов, выполняющих одни и те же функции, может быть несколько – и каждый из них будет ответственен за свой фронт работ. Для примера рассмотрим следующую ситуацию. Большая холдинговая компания внедряет у себя систему дистанционного обучения. У компании в разных регионах много филиалов, в каждом из которых существует свой учебный центр. То есть, планирование обучения и оценка результатов происходит на местах. В этом случае хорошо бы в каждом филиале иметь своего администратора системы, который сможет самостоятельно назначать электронные курсы и тесты сотрудникам и контролировать результаты. Но при этом такой администратор сможет видеть только сотрудников из своего филиала и работать только с ними. И все же это будет единая система, так что главный администратор сможет уже управлять всеми сотрудниками из всех филиалов, а также контролировать деятельность региональных администраторов.

Таким образом, система должна позволять распределенное управление. В системе должна поддерживаться работа неограниченного количества администраторов с распределенными правами доступа к тому или иному функционалу администрирования.



## Заключение

В этом небольшом сборнике рассмотрены основные критерии, на которые стоит обращать внимание при выборе системы дистанционного обучения. Конечно, этот перечень не претендует на полноту и завершенность. Это лишь частное мнение. Кто-то данный список либо дополнит своими требованиями, либо сократит.

Кроме перечисленного выше, при выборе системы можно также обращать внимание на следующие факторы:

- Насколько удобен и понятен интерфейс;
- Насколько проста система в установке и настройке;
- С какими базами данных может работать система;
- Какова архитектура системы;
- Какой поддерживается функционал сайта – форумы, чаты, блоги, голосования, управление баннерами и пр.;
- Есть ли решения для работы с неудовлетворительными каналами связи или с полным их отсутствием в удаленных точках;
- Насколько поддерживается работа системы с мобильными устройствами и так далее.

Требований может быть много, для каждого заказчика свои в зависимости от целей внедрения системы. Однако хочется верить, те, которые были сформулированы выше, так же окажутся полезными, и позволят сориентироваться в море программного обеспечения.

Свои объективные отзывы, искренние пожелания и гневную критику отправляйте по адресу: [book@kursor.com.ru](mailto:book@kursor.com.ru).

