

рекламу, поскольку: эти нормы не учитывают технические особенности сети Интернет; затруднено определение юрисдикции (законодательство какой территории применять); затруднен контроль и привлечение к ответственности нарушителей.

Таким образом, законодательные меры борьбы со спамом должны иметь комплексный характер и не сводиться лишь к юридическим запретам и санкциям. Поскольку спам может рассматриваться как реклама, соответствующие сообщения должны полностью соответствовать законодательству о рекламе, в частности Республики Беларусь. Но должны быть и дополнения, учитывающие электронный способ рассылки таких сообщений. Например, необходимо указывать в заголовке сообщения, что оно носит рекламный характер. Должны быть регламентированы и права абонентов – владельцев электронных адресов и др.

В последние годы в законодательство многих стран появляются специализированные нормы, регулирующие рекламу в Интернете. В ряде европейских стран и некоторых штатах США были приняты законы позволяющие привлекать к ответственности спамеров уже за сам факт рассылки ими не запрошенных адресатами сообщений. Например, законопроект по борьбе со спамом Сената США квалифицирует рассылку как незаконную, «если ее получатель не выразил на то своего желания». Закон запрещает спамерам скрывать свои почтовые адреса, брать адреса для рассылки с веб-сайтов и оставлять пустым поле для темы посылаемого сообщения.

При этом специалисты признают, что технологически пока нет действующих механизмов для поимки спамеров – используемые ими виртуальные серверы управляются дистанционно и постоянно меняют свое расположение.

Следующим важным направлением борьбы со спамом является применение технических средств защиты, в частности программных средств, позволяющих отфильтровывать и удалять «подозрительные сообщения» еще до получения их адресатом.

Еще одним направлением борьбы со спамом является проведение разъяснительной работы о его противоправности и о необходимости борьбы с ним. Например, необходимо следовать рекомендациям: никогда не оставлять своего электронного адреса незнакомым людям; не отвечать на сообщения, содержащие спам; быть разборчивым в выборе провайдера услуг доступа и обращать внимание на пункты абонентского договора и др.

Список использованных источников:

1. Якушев, М. А. Четыре способа остановить спам: разъяснять, программировать, принимать законы и штрафовать // Независимая газета. – 2003. – 20 июня.

2. Юрасов, А.В. Основы электронной коммерции / А.В. Юрасов – М. : Телеком, 2008. – 480 с.

3. Кодекс Республики Беларусь об административных правонарушениях : принят Палатой представителей 17 дек. 2002 г. : одобр. Советом Респ. 2 апр. 2003 г. : с изм. и доп. по состоянию на 08.01.2018 г. – Минск : Нац. центр правовой информ. Респ. Беларусь, 2018. – 301 с.

4. О рекламе : Закон Респ. Беларусь, 10 мая 2007 г. в ред. Закона Респ. Беларусь от 17.07.2017 г. // Консультант Плюс : Беларусь. Технология 3000 [Электронный ресурс] / ООО «ЮрСпектр», Нац. центр правовой информ. Респ. Беларусь. – Минск, 2018.

IT КОМПАНИИ: ОРГАНИЗАЦИЯ ТРУДОВОГО ПРОЦЕССА

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Кособудский А.В.

Ермакова Е.В. – к.э.н., доцент

Организация труда — это конкретные формы и методы соединения людей и техники в процессе труда с целью достижения полезного эффекта трудовой деятельности. Т.к. основным продуктом деятельности IT- компании является готовый программный продукт, рассмотрим примеры взаимодействия трудящихся при различных моделях и методологиях разработки программного обеспечения.

Разработка программного продукта знает много достойных методов и методологий — иначе говоря, устоявшихся «best practices». Выбор зависит от специфики проекта, системы бюджетирования и субъективных предпочтений руководителя [1]. Рассмотрим наиболее известные.

Каскадная модель (модель «Водопад») — модель процесса разработки программного обеспечения, в которой процесс разработки выглядит как поток, последовательно проходящий фазы анализа требований, проектирования, реализации, тестирования, интеграции и поддержки. Следуя каскадной модели, разработчик переходит от одной стадии к другой строго последовательно. Сначала полностью завершается этап «определение требований», в результате чего получается список требований к ПО. После того как требования полностью определены, происходит переход к проектированию, в ходе которого создаются документы, подробно описывающие для программистов способ и план реализации указанных требований. После того, как проектирование полностью выполнено, программистами выполняется реализация полученного проекта. На следующей стадии процесса происходит интеграция отдельных компонентов, разрабатываемых различными командами программистов. После того, как реализация и интеграция заверше-

ны, производится тестирование и отладка продукта; на этой стадии устраняются все недочёты, появившиеся на предыдущих стадиях разработки. После этого программный продукт внедряется и обеспечивается его поддержка — внесение новой функциональности и устранение ошибок. Тем самым, каскадная модель подразумевает, что переход от одной фазы разработки к другой происходит только после полного и успешного завершения предыдущей фазы, и что переходов назад либо вперёд или перекрытия фаз — не происходит.

«V-Model»- унаследовала структуру «шаг за шагом» от каскадной модели. V-образная модель применима к системам, которым особенно важно бесперебойное функционирование. Например, прикладные программы в клиниках для наблюдения за пациентами, интегрированное ПО для механизмов управления аварийными подушками безопасности в транспортных средствах и так далее. Особенностью модели можно считать то, что она направлена на тщательную проверку и тестирование продукта, находящегося уже на первоначальных стадиях проектирования. Стадия тестирования проводится одновременно с соответствующей стадией разработки, например, во время кодирования пишутся модульные тесты.

Итерационная модель - не требует для начала полной спецификации требований. Вместо этого, создание начинается с реализации части функционала, становящейся базой для определения дальнейших требований. Этот процесс повторяется. Версия может быть неидеальна, главное, чтобы она работала. Понимая конечную цель, команда разработки стремится к ней так, чтобы каждый шаг был результативен, а каждая версия — работоспособна [2].

Многие из организаций, занимающихся разработкой программного обеспечения, пытаются сделать свои процессы разработки более гибкими. В феврале 2001 в штате Юта США был выпущен «Манифест гибкой методологии разработки программного обеспечения» - AGILE [3].

Agile — семейство процессов разработки, а не единственный подход в разработке программного обеспечения. Agile не включает практик, а определяет ценности и принципы, которыми руководствуются успешные команды. Выделяют основные идеи Agile:

- люди и взаимодействие важнее процессов и инструментов;
- работающий продукт важнее исчерпывающей документации;
- сотрудничество с заказчиком важнее согласования условий контракта;
- готовность к изменениям важнее следования первоначальному плану [4].

В «гибкой» методологии разработки после каждой итерации заказчик может наблюдать результат и понимать, удовлетворяет он его или нет. Это одно из преимуществ гибкой модели. К ее недостаткам относятся то, что из-за отсутствия конкретных формулировок результатов сложно оценить трудозатраты и стоимость, требуемые на разработку. Экстремальное программирование (XP), SCRUM, Kanban, RUP, CMM являются одними из наиболее известных применений гибких моделей на практике.

Такие гибкие методологии подходят для больших или нацеленных на длительный жизненный цикл проектов, постоянно адаптируемых к условиям рынка [5].

Список использованных источников:

- Рухляда И.В. Организация труда в компаниях, разрабатывающих программное обеспечение // Труд за рубежом. 2009. - № 3.
- Руководство к своду знаний по управлению проектами.: руководство PMBOK / Project Management Institute. г. Нью-тон-Сквер, Пенсильвания, США : PMI, Inc, 2008. - 494 с.
<http://agilemanifesto.org>
- Соммервилл И. Инженерия программного обеспечения. 6-е изд. - М.: Вильямс, 2002. - 623 с.
- Кон М. Scrum. Гибкая разработка ПО . — М. : Вильямс, 2011.576 с.

5S – СИСТЕМА ОРГАНИЗАЦИИ И РАЦИОНАЛИЗАЦИИ РАБОЧЕГО МЕСТА В IT-КОМПАНИИ

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Куликовская А.В., Лютак В.В., Лубневская А.В.

Горноста́й Л.Ч. – ст. преподаватель

Работники различных служб, в подавляющем большинстве, являются офисными сотрудниками. Среди сотрудников IT-компаний фактор комфортности рабочего места ценится выше, чем наличие социального пакета, престижность компании, местоположение офиса, так как там они проводят весь свой рабочий день. Большинство работников считают, что эффективность их труда напрямую зависит от эргономичности и удобства рабочего места. Для повышения производительности труда и комфорта работников используется 5s-система.

Система 5S - это система организации рабочего места, которая позволяет значительно повысить эффективность и управляемость операционной зоны, улучшить корпоративную культуру, повысить производительность труда и сохранить время [1].

В систему 5S входят пять действий:

1) Сортировка - означает, что вы высвобождаете рабочее место от всего, что не понадобится при выполнении текущих производственных операций;