

ВЗАИМОДЕЙСТВИЕ РАЗРАБОТЧИКА И ПОЛЬЗОВАТЕЛЕЙ МЕДИЦИНСКОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ НА ЭТАПЕ ЕЁ ЭКСПЛУАТАЦИИ

СОДЕРЖАНИЕ:

Введение.

1. ЗАПРОСЫ К РАЗРАБОТЧИКУ.

- 1.1. Причины и значение.
- 1.2. Содействие и организация.
- 1.3. Виды и формы.
 - 1.3.1. Сообщения об аварийных завершениях и зависаниях программы.
 - 1.3.2. Сообщения о нелогичных действиях программы или неверном расчёте.
 - 1.3.3. Сообщения об ошибках в текстах.
 - 1.3.4. Запросы на пополнение справочников.
 - 1.3.5. Совершенствование функций, новые функции, изменения в структурах.
 - 1.3.6. Замечания о дизайне и интерфейсе.
- 1.4. Основания для отклонения запроса.
- 1.5. Передача запросов.

2. УДОВЛЕТВОРЕНИЕ ЗАПРОСОВ ПОЛЬЗОВАТЕЛЕЙ.

- 2.1. Способы, инструменты, специалисты.
 - 2.1.1. Общие условия.
 - 2.1.2. Особенности дополнения системных справочников.
 - 2.1.3. О соотношении между системными справочниками и официальными классификаторами.
 - 2.1.4. О конструкторах форм.
 - 2.1.5. О вводе текстовых шаблонов
 - 2.1.6. О консультантах.
- 2.2. Сроки удовлетворения запросов

3. ПРАВОВЫЕ, МОРАЛЬНЫЕ И ФИНАНСОВЫЕ АСПЕКТЫ ПОДДЕРЖКИ МИС.

- 3.1. Правовые и моральные аспекты.
- 3.2. Затраты разработчика на поддержку МИС.
- 3.3. Сколько стоит поддержка МИС для ЛПУ.

ЗАКЛЮЧЕНИЕ. ВЗАИМОДЕЙСТВИЕ РАЗРАБОТЧИКА И ПОЛЬЗОВАТЕЛЕЙ МИС – АНТИЭНТРОПИЙНАЯ СИСТЕМА.

ВВЕДЕНИЕ

Переход лечебно-профилактического учреждения (ЛПУ) на использование любой новой медицинской информационной системы (МИС) осуществляется в 3 этапа:

- 1) выбор системы, создание условий для ее внедрения, поставка системы,
- 2) ввод в эксплуатацию и доработка на этапе ввода,
- 3) производственная эксплуатация.

Переход на использование МИС означает для всего ЛПУ радикальное преобразование повседневной работы с информацией. Оно охватывает всех участников лечебно-диагностического процесса (ЛДП): врачей, медсестёр, старших медсестёр, работников лабораторий и специализированных кабинетов, заведующих отделениями и параклиническими подразделениями, медицинских статистиков, главного врача и его заместителей, студентов, курсантов и преподавателей медицинских вузов, научных работников, а также внешних контролёров и экспертов. Привычная, гибкая и всё терпящая бумажная технология почти полностью вытесняется компьютерной. Вытесняется раз и навсегда, необратимо. Мы имеем дело с чрезвычайно ответственной ситуацией.

На этапе ввода МИС осуществляется ее первоначальная адаптация, которая требует максимальных усилий с обеих сторон: и от пользователей, и от разработчика. Пользователи вынуждены освоить новые приёмы и правила, выработать новые привычки, изжить старые. Разработчик же, наблюдая поведение своего детища в настоящей жизни, должен на ходу улавливать обнаруживаемые на первых порах неудачные решения и упущения и тут же, как можно быстрее, устранять всё, что мешает медикам вернуться к спокойному выполнению своей работы с пациентами.

В процессе ввода заслуживают конструктивной реакции все без исключения замечания, жалобы, претензии пользователей, кто бы и как бы их ни высказывал. Не имеет значения, в чём и в ком их причина: в упущениях программиста, в том, что медики с самого начала недостаточно ориентировали разработчика или в непонимании, в неумении, в страхе пользователей перед техникой, в самой технике. Врачи не виноваты, что их выхватили из одной среды и погрузили в другую, им надо работать, за ними – пациенты. Нельзя требовать, чтобы они детально разбирались в происхождении обнаруживаемых неудобств. В конце концов, для них все вдруг возникшие помехи прямо ли, косвенно ли спровоцированы разработчиком. Ему и отвечать. Как можно раньше, как можно терпеливее, привлекая при необходимости техников, системного администратора, главного врача. Есть, конечно, на этот счёт общепринятые подходы, международные рекомендации, да и условия

контрактов на поставку и внедрение МИС прописываются не для того, чтобы заполнить пустые страницы в договоре. Но, тем не менее, разбираться во всем этом следует непосредственно разработчику (а если внедрение осуществляет другая организация, то ей) совместно с руководителем проекта автоматизации со стороны ЛПУ. Пользователям же целесообразно предоставить полную свободу выражения. До поры, до времени.

На то, чтобы основная волна животрепещущих проблем схлынула, чтобы врачи стали уверенно вести электронные медицинские карты (ЭМК) или электронные истории болезни (ЭИБ), медицинские статистики - составлять отчёты, а руководители – пользоваться всей полнотой информации для управления лечебно-диагностическим процессом, обычно требуется от 4 недель до 3 месяцев. Этим и заканчивается этап ввода МИС в эксплуатацию. Далее пойдёт её спокойное использование, когда система стала уже привычной, обыденной, когда врачи о ней уже не думают, как раньше не думали о ручке и бумаге.

Однако, в отличие от ручки и бумаги, МИС время от времени даёт знать о своём неизбежном несовершенстве. На этапе внедрения выявляются не все её недостатки, а только те, которые на коротком отрезке времени встречаются неоднократно. В тех же функциях, к которым врачи прибегают редко, ошибки программиста могут не выявляться годами, а потом вдруг обнаружатся. Программисты хорошо знают, что часть ошибок возникает только при некоторых сочетаниях условий во время работы программы, а все сочетания в многосложной деятельности врача заранее не предусмотреть. И с течением времени они нет-нет да и возникают, провоцируя неправильное поведение информационной системы.

Впрочем, на этапе производственной эксплуатации, о котором идёт здесь речь, устранение ошибок и упущений разработчика – не единственный предмет взаимоотношений сторон. Другой фактор, требующий взаимодействия пользователей с разработчиком – это постоянное развитие медицинской науки и организации лечебного дела, вследствие которого в медицинской практике появляются новые понятия, новые средства, новые правила, и всё это выражается в информации, в работе с ней. Сверх этих внутренних процессов роста, на ЛПУ оказывается воздействие и извне, со стороны органов управления здравоохранением и субъектов медицинского страхования. У них свои запросы на информацию, свои требования к документации, довольно часто меняющиеся.

Если количество несовершенств, изначально допущенных программистом, со временем стремится к нулю, то развитие ЛПУ и изменения внешних требований бесконечны. Поэтому МИС всегда будет испытывать хотя бы кратковременные отставания от требований времени то в одном пункте, то в другом. Все это вместе взятое будет требовать постоянных изменений в МИС. Реализовывать эти изменения может только разработчик. Но и у пользователей есть важная обязанность – как можно раньше указывать разработчику на

недостатки и как можно точнее их описывать. Если таких усилий с обеих сторон не будет, МИС начнёт хромать, в ней станут накапливаться помехи, её работоспособность будет неуклонно снижаться, а неудовлетворённость врачей системой – нарастать.

При этом для ЛПУ будет утрачиваться одно из важнейших достоинств современных информационных технологий – эффект быстрого обучения целых коллективов. Новшества медицины и организации, большие и малые, - это шаги прогресса только тогда, когда они осваиваются всеми врачами. При традиционной организации (при бумажной технологии) с ними знакомятся, усваивают их, используют в своей практике сначала лишь немногие - большинство ещё долго следует своим привычкам. В условиях МИС всё новое предлагается сразу всем, система напоминает о нём, позволяет контролировать его использование, а в части случаев делает старые приёмы просто невозможными. Значит, пользователи всегда будут нуждаться в том, чтобы в МИС вносились дополнения и усовершенствования.

Итак, взаимодействие пользователей и разработчика – процесс бесконечный и равно необходимый обеим сторонам: врачам нужен всегда современный и эффективный инструмент, разработчику – уверенность в высоком качестве его продукта, которое является обязательным условием успешности работы на рынке. Поэтому такое взаимодействие заслуживает пристального внимания, тем более, что отношения здесь не сводятся к формуле «скажите – сделаем», они намного сложнее.

1. ЗАПРОСЫ К РАЗРАБОТЧИКУ

1.1.. Причины и значение.

МИС нужна для рационального использования информации. Запросы к разработчику возникают у пользователя тогда, когда при работе с информацией он испытывает неудобства или вообще не может с помощью системы выполнить необходимую ему операцию. Кроме того, по мере привыкания к системе пользователи обнаруживают, что те или иные функции сделаны, на его взгляд, не лучшим образом. Без этого не обходится ни одна МИС: разработка обширных программ – это труд длительный, обычно коллективный и далеко не всегда подчинённый строгому единому руководству. Поэтому квалификация разработчиков растёт по ходу создания продукта, а в результате со временем пытливые пользователи неизбежно приходят к двум видам критических оценок. Во-первых, они сравнивают условия работы в своей МИС с другими известными им программными продуктами и начинают понимать, что кое-что у последних было бы неплохо позаимствовать. Во-вторых, они волей-неволей обнаруживают, что почему-то разработчик некоторые сходные по форме операции осуществляет различно, хотя для технических процедур предпочтительна однотипность, особенно если не все варианты одинаково рациональны.

Итак, причина запросов – неизбежное несовершенство МИС. Сначала это исключительно недоработки её создателей: постановщиков задач, разработчиков, руководителя проекта, которые непременно обнаруживаются по мере того, как пользователи всё лучше вникают в свойства МИС, всё полнее их используют, всё больше отходят от прежних способов работы с информацией. Этот поток запросов тем больше, чем менее совершенен изначальный продукт, По мере роста активного отношения пользователей он некоторое время нарастает, но потом рано или поздно иссякает. Зато появляются запросы, вызванные переменами в самой медицине и в организации лечебного дела. Предвидеть их никто не может, понемногу такие перемены совершаются постоянно, но бывают и радикальные перестройки организации, и скачкообразное развитие в той или иной области медицины, - тогда число запросов может резко возрасть, а сами запросы усложняются. Как бы ни было, речь всегда идёт о необходимом совершенствовании важнейшего инструмента работы со сведениями о пациентах и деятельности врачей, о качестве той стороны лечебно-диагностического процесса, которая зависит от прежних, нынешних или будущих усилий разработчика.

Для разработчика запросы пользователей – основной надёжный источник информации о его детище. Они практически никогда не бывают напрасными, так как порожда-

ются не умозрительными соображениями, не сопоставлением устройства МИС с теми или иными выдвинутыми на сегодня принципами или правилами, а реальными ситуациями в практике медика, приложением информационной технологии к текущему производственному процессу.

Конечно, всё начинается с выражения общего недовольства и аморфных претензий: («нам не нравится», «сделайте удобнее», «добавьте такую-то возможность»). Но тут должны действовать строгие организационные правила: системный администратор, сам разработчик или руководитель ЛДП (заведующий отделением, главный врач, начмед) используют каждое такое высказывание для того, чтобы потребовать от автора замечания текст с описанием проблемы и демонстрацию её на экране или в напечатанной выходной форме. Только тогда выражение недовольства превращается в действенный запрос, в противном случае оно расценивается как мешающее работе брюзжание, не заслуживающее внимания.

Разумеется, предельная конкретность замечаний побуждает к обратному процессу: их всегда надо со знанием дела обобщать, экстраполировать на другие сходные ситуации, но они никогда не беспочвенны. Это всегда сигналы о действительном неблагополучии, которое, как правило, можно и нужно устранить с пользой для множества людей. В том числе и с пользой для репутации МИС и её создателей.

Запрос к разработчику должен быть достаточно информативным, чтобы тот в деталях понял, что ему делать. Не все пожелания врачей бесспорны - иные надо специально обосновывать. Значит, разработчик должен относиться к запросам критически, в ряде случаев требовать аргументации, уточнений, привлекать экспертов. Всегда возможны противоречивые запросы: одно и то же разные врачи могут формулировать по-разному, не с каждым дополнением или изменением всегда согласятся все. Наконец, далеко не каждому врачу свойственно системное мышление, понимание, что предлагаемые им изменения могут отразиться сразу на многих функциях и что это надо непременно просчитать. Всё это означает, что взаимодействие разработчика и пользователя требует определённой организации, правил, распределения обязанностей и прав, специального инструментария. При этом от обеих сторон требуется понимание общих целей поддержки МИС разработчиком. В этом смысле термин «поддержка» надо оберегать от эпитета «техническая», потому что забота о техническом совершенстве системы – только самая простая и не главная часть задачи. Её главная часть – это поддержка на должном уровне медицинского содержания системы.

Можно утверждать, что без учета запросов пользователей именно в ходе производственной эксплуатации, в обстановке обычной спокойной работы, при постепенном осво-

ении новой технологии на всю её глубину нельзя создать хорошую МИС. Информационные процессы, идущие в лечебном учреждении, сложны, многообразны и на исходной позиции так мало формализованы, что никакой постановщик задач, никакой разработчик, будь он хоть врачом, хоть программистом, хоть тем и другим в одном лице, не может предусмотреть все детали. Все детали выявляются пользователями: постепенно, в ходе эксплуатации на протяжении месяцев и лет. И так же постепенно в ходе эксплуатации нарастает качество МИС.

МИС может страдать множеством недостатков, но при рационально организованном взаимодействии разработчика с пользователями они шаг за шагом устраняются. Наоборот, без такого взаимодействия самая продуманная и развитая МИС через некоторое время будет восприниматься врачами как инструмент неполноценный, неудобный, неперспективный.

1.2. Содействие и организация.

Итак, запросы к разработчику необходимы обеим сторонам, необходимы как постоянный компонент жизни и развития информационной системы. Надо подчеркнуть, что речь идёт именно о запросах, основанных на личной работе в системе, а не о замечаниях созерцателей, которым со стороны показалось, будто что-то можно сделать лучше. Глядя со стороны, можно рассуждать только об общих принципах, а для разработчика это неконструктивный подход. Для этапа производственной эксплуатации как нельзя более подходит правило «Принципы – ложь, истина – в оттенках». Тут надо улучшать практическое осуществление принятых ранее принципов. Например, бессмысленно вообще повторять, что МИС должна соответствовать правилу «однократный ввод - многократное использование»: это мало что даёт разработчику, под таким лозунгом можно предлагать что угодно. Вместо этого надо делать конкретные предложения об упрощении ввода данных в определённом месте электронного документа.

Кроме того, замечания со стороны не обязывают к ответственности. Отреагировав на них, часто уже не у кого спросить, достаточно ли хорошим получился результат. А ведь взаимодействие предполагает, что тот, кто о чём-то попросил и потом получил, должен подтвердить своё удовлетворение или, увидев, что получилось не совсем то, уточнить свои предложения. Это циклический процесс и удовлетворение запроса нередко требует нескольких итераций, каждый раз всё больше детализирующих желательные изменения. Участвовать в таком процессе может только тот, кто работает в системе, кто заботится о её совершенствовании для самого себя. Это не значит, что не надо прислушиваться к суж-

дениям резонёров, просто их замечания ни к чему не обязывают разработчика, тогда как замечания пользователей всегда требуют полного внимания и конструктивного ответа.

Вообще-то, делать кому-нибудь замечания – дело непростое изначально, особенно если речь идёт не о безадресном брюзжании, а о том, что надо потратиться на описание конкретных деталей и приведение примеров. Как правило, пользователи терпеливы, тем более, что в большинстве своём они не имеют достаточного представления о том, можно или нет что-то изменять в системе, насколько это сложно. Если к встретившимся проблемам можно как-то приспособиться, как-то их обойти, тем более, если остаётся возможность то тут, то там воспользоваться вместо программных процедур текстовыми записями, а то и просто вернуться к ручке и бумаге, именно это и будет сделано. И информация о недостатках системы к разработчику не придёт.

Чтобы для пользователей стало правилом не приспособление к неудобству, а требование его устранить, надо обеспечить тем, кто хочет сообщить о своей неудовлетворённости, предельную простоту действий, создать специальные условия. Международной практикой и рекомендациями (например, ИТIL) такие мероприятия принято называть организацией службы Service desk (или Helpdesk). Эта служба должна регистрировать обращения пользователей, какие бы они ни были, ранжировать их по видам (ошибки, вопросы по работе, предложения по улучшению МИС и т.д.) и обеспечивать их передачу тем, кто должен обеспечить устранение выявленных дефектов.

Информационная система, отвечающая за регистрацию обращений и контроль за их удовлетворением, может быть как частью самой МИС, так и внешним приложением. Последних сегодня множество – как больших коммерческих продуктов, так и простых в использовании бесплатных OpenSource-решений. Наличие такой системы при внедрении и эксплуатации МИС - неременное условие. Она должна быть доступна на каждом автоматизированном рабочем месте. В неё пользователь должен иметь возможность вписать свои замечания и просьбы, будучи уверен, что они попадут в поле зрения главного врача, системного администратора и, при необходимости, будут переданы разработчику, а главное – что на них будет эффективная реакция или вразумительный ответ.

Правильная организация работы такого канала информации предполагает следующее. Введенная информация должна быть снабжена датой ввода и поступать в оперативные сводки (в поликлинике – еженедельные, в стационаре – ежедневные) до тех пор, пока тот, кто ввёл запрос в систему, сам не отметит, что он удовлетворён результатом. Включение информации о запросах в состав сводок позволяет главному врачу следить и за активностью врачей, и за содержанием проблем, и за сроками их разрешения. Заведующие отделениями на основании тех же сводок могут своевременно корректировать запросы,

уточнять их или в случаях, когда запрос оказывается результатом недоразумения, брать разрешение проблемы на себя.

Роль непосредственного адресата запросов следует возложить на системного администратора. Пользователь не должен заботиться о том, кто именно обязан обеспечить ему устранение встретившихся недостатков, будь то непонятные действия программы, орфографические ошибки на экране, неисправность принтера или нехватка бумаги. Другое дело – сисадмин: он способен и должен отделить то, что находится в сфере ответственности технических и хозяйственных служб или в его собственной, от информации, которую надо передать разработчику.

Следующая задача системного администратора – вникнуть в содержание запроса, проверить его обоснованность, то есть самому или вместе с автором постараться увидеть или спровоцировать ту самую ситуацию, которая вызывала неудовлетворённость пользователя. При этом в части случаев выясняется, что дело было просто в неверных действиях (нажали не ту клавишу, не обратили внимания на рекомендации на экране и проч.), - тогда всё разрешается тут же. Знакомство же с проблемой, действительно требующей передачи разработчику, позволяет сисадмину внести в запрос все сведения, которые тому требуются.

Разработчик, в свою очередь, должен заранее оговорить с ЛПУ, что именно и в каком виде надо ему сообщать в зависимости от вида запроса. Общепринятой мировой практикой является регламентация такой работы с помощью специального «Соглашения об уровне предоставления услуги» (англ. Service Level Agreement - SLA).

1.3. Виды и формы.

Проблемы работы в МИС, с которыми сталкивается пользователь, разнообразны и неравнозначны. Одни требуют немедленного решения, поскольку вообще не позволяют выполнять те или иные функции, с другими можно повременить. Одни разработчику сразу понятны и могут быть устранены в считанные минуты, другие требуют значительных программистских усилий, третьи – предварительного тщательного обдумывания и обсуждения с авторами запросов. Одни являются результатом недосмотра, недоработок создателей МИС, другие – следствием развития лечебного учреждения или появления новых требований в медицине и организации здравоохранения, третьи диктуются индивидуальной особенностью данного ЛПУ или региональной организации системы здравоохранения. Всё это имеет значение для принятия решений в разных ситуациях, но первое и самое важное для взаимодействия пользователей с разработчиком – это разделение запросов по форме, в которой они должны быть представлены, чтобы быть для разработчика доста-

точно информативными. Тут пользователю должны быть заданы правила, за соблюдением которых обязан следить системный администратор. Попробуем классифицировать запросы по этому принципу.

1.3.1. Сообщения об аварийных завершениях и зависаниях программы.

От проблем этого рода не застрахован никто, несмотря на тестирование до внедрения и все поправки на этапе ввода. Заранее всё протестировать невозможно, некоторые сочетания условий возникают редко, к иным функциям, предусмотренным в программе, пользователи могут долго не прибегать, но всё плохое, что может случиться, обязательно однажды случится и потом будет повторяться. При зависании программы в определённый момент работы надо сообщить программисту последовательность действий, ведущих к этому моменту. Сообщить протокольно, точно называя использованные пункты экрана и порядок действий пользователя. Если речь идёт о работе с конкретной историей болезни, надо её проанализировать. Разработчику обычно требуется повторить такую ситуацию у себя, системный администратор должен это понимать, сам всё увидеть и позаботиться о точности описания.

То же самое следует сказать об аварийном завершении программы или работы ее отдельных функций. Но тут надо добавить, что в самой программе должна быть заложена функция обработки ошибок, которая, во-первых, записывает в файл ошибок всю необходимую для разработчика информацию, а, во-вторых, выдаёт на экран рекомендации пользователю (известить системного администратора, повторить процедуру и проч.). Таким образом, разработчику передаётся не только протокол действий пользователя и указание на историю болезни, но и файл ошибок.

Если в МИС не предусмотрена функция обработки аварийных ситуаций с записью в специальный файл или журнал работы МИС, это принципиальный дефект, который необходимо устранить.

Так как зависания и аварийные завершения возникают вследствие обработки определённых данных, разработчику для понимания причин необходимы именно эти данные. Следовательно, у него должен быть доступ к этой базе данных, либо ему надо передавать её копию.

1.3.2. Сообщения о нелогичных действиях программы или неверном расчёте.

Проблемы этого вида обычно связаны с ошибками в алгоритме программы, которые, в свою очередь, имеют причиной либо простую погрешность программиста, либо неверную или недостаточно полную постановку задачи. Разработчику в таких случаях тоже

нужны совершенно конкретные примеры, документированные копиями экранов, и возможность самому вызвать ту же самую ситуацию. Если речь идет о неверном соблюдении каких-либо нормативных актов (например, неправильный расчет в государственном статистическом отчете или неправильный вывод утвержденной учетной формы), то вместе с описанием проблемы разработчику должна быть предоставлена и ссылка на соответствующий нормативный документ, который устанавливает правильный порядок расчетов, внешний вид документа и т.д.

1.3.3. Сообщения об ошибках в текстах.

Огромное количество текстов в МИС (экранные элементы, содержание справочников, текстовые шаблоны) не обходится без ошибок и опечаток. Все они вылавливаются только с помощью придирчивых врачей. Разработчику надо сообщить, о каком именно объекте идёт речь, повторить грамматически или синтаксически неправильное написание, привести правильное написание и приложить копию экрана, на котором видна ошибка. Заметим, что, если ошибка допущена в справочнике, сисадмин должен понимать, когда речь идёт о системном справочнике, за который в ответе именно разработчик, а когда о местном, который составляется и корректируется в самом ЛПУ.

1.3.4. Запросы на пополнение справочников.

Прежде всего, определимся в понятиях. Такие справочники, как «Сотрудники», «Подразделения», «Медицинские учреждения» и др., служат для описания местных обстоятельств медицинской деятельности. Остальные нужны, чтобы описывать состояние пациента и действия врача, их содержание – это общая для всех медицина, они должны быть одинаковыми во всех учреждениях, где используется данная МИС. Первые назовём местными, вторые – системными (ещё их называют соответственно локальными и глобальными). Врачам различать их не нужно, а системному администратору необходимо. Обязанности разработчика по отношению к местным справочникам исчерпываются созданием удобного инструмента для работы с соответствующей информацией, системные же справочники нуждаются в его постоянной поддержке.

Системные справочники, а также все виды текстовых шаблонов – это медицинское содержание системы и, соответственно, её медицинская культура. Чтобы содержание было полноценным, а уровень культуры высоким во всех учреждениях, эти инструменты МИС должны поддерживаться только централизованно. Идею передать их пополнение и коррекцию самим пользователям надо сразу отклонить как принципиально порочную. Она означает, что через некоторое время разные учреждения, пользующиеся вроде бы одной и

той же МИС, окажутся в разной информационной среде, не говоря уж о том, что необходимое качество справочников системы при таком порядке работы гарантировать невозможно.

У каждого системного справочника свои особенности. Минимальный вариант структуры справочника – наименование и код. Пользователь сообщает, какое наименование надо добавить, разработчик присваивает ему код. Это – в большинстве случаев. Но несколько справочников содержат и другие структурные элементы, информацию для которых разработчик должен получить либо от пользователя, либо из надёжной литературы. Укажем на них:

Справочник диагнозов. В этом качестве сегодня, как правило, используется Международная классификация болезней 10-го пересмотра. Так как официальная отчётность основана именно на этой классификации, она вполне удовлетворяет медицинских статистиков. Но лечащему врачу, описывающему пациента, в отдельных её разделах недостаёт детализации. Это и порождает запросы. Их несложно удовлетворить, если воспользоваться оставленной в МКБ-10 возможностью делать подрубрики, ничуть не нарушая общей конструкции.

Так, например, эндокринологов не удовлетворяет только общая формулировка «E23.33 Синдром пустого турецкого седла», так как этот диагноз в истории болезни надо характеризовать наличием тех или иных важных проявлений: ожирением, гиперпролактинемией, нарушениями менструального цикла и других. Общее число таких синдромов плюс вариант, когда у пациента есть их комбинация, - десять. Отразить имеющееся состояние, можно, конечно, с помощью текстовой записи, но это означает, во-первых, лишний труд, а, во-вторых, невозможность последующих выборок по важным признакам. Между тем, ничто не мешает к коду E23.33 добавить 10 подрубрик от E23.330 до E23.339 и пользоваться всеми преимуществами формализованной информации. Понятно, что при этом все эти диагнозы в официальном отчёте попадут в более старшую рубрику E23.33.

Подобные проблемы в разных местах МКБ возникают у ревматологов, кардиологов, сосудистых хирургов, офтальмологов, инфекционистов, пульмонологов, травматологов, онкологов, психиатров. Все они достаточно изящно решаются, но важно, чтобы пользователь, не только потребовал, чтобы справочник пополнился той деталью, которой ему не хватает сегодня, а взглянул на задачу чуть шире и перечислил все возможные варианты детализации этого самого диагноза.

Справочник средств лечения. Это – одна из самых динамичных структур МИС. Средства лечения, медикаментозного, физиотерапевтического, психотерапевтического, варианты диет и лечебных режимов совершенствуются постоянно. Чтобы врач мог назна-

чать их в истории болезни, они должны быть в справочнике. Если чего-то там нет, врач должен потребовать дополнения, указав точное наименование медикамента, диеты, лечебного режима, физиотерапевтической процедуры, психотерапевтического тренинга. Здесь важен ряд деталей.

Физио- и психотерапевтические процедуры должны быть поименованы полностью, без сокращений, так, чтобы смысл был доступен не только специалисту в определённой области, а каждому медику (в том числе главному врачу, его заместителю, внешним контролёрам и экспертам). Далее, предполагается, что справочник средств лечения содержит раздел для кратких комментариев, и эти комментарии пользователь должен непременно дать, если речь идёт о новой диете, новом лечебном режиме или психотерапевтическом тренинге: (имеются в виду сведения о сути метода, показаниях и противопоказаниях к нему).

Отсутствие в справочнике назначаемых врачом медикаментов – частая причина запросов в ходе производственной эксплуатации МИС. Здесь требования к форме запроса могут и должны быть минимальными: достаточно всего лишь сообщить наименование медикамента. Всё остальное разработчик узнает сам, благо заслуживающие доверия источники (справочник Видаль и энциклопедия Российские Лекарственные Средства) общедоступны. Из этих источников он сам выберет и введёт в справочник все варианты использования лекарства (лекарственные формы, пути введения, разовые дозы, которые должны появляться в назначениях врача), а также краткие комментарии о показаниях, противопоказаниях и высших дозах. Попытки получать эти сведения от самих врачей надо признать заведомо бесперспективными: это обязательно обернётся неполнотой сведений, дубликатами и частыми ошибками, которые здесь недопустимы. В случаях, когда медикамент в справочнике есть, но нет строки с необходимой дозой или с необходимой лекарственной формой (например, есть таблетки, а нужны капли, свечи, инъекции), пользователь, конечно должен об этом сообщить.

Справочник сигнатур. Предполагается, что ритм применения лечебного средства врач назначает с помощью отдельного справочника сигнатур. Этот ритм нельзя назначать путём дописывания с клавиатуры, потому что тогда не будет возможности программно подсчитывать количество принятых медикаментов и стоимость лечения. Время от времени оказывается, что в этот справочник надо внести неожиданные добавления, например, такие: «по 1/4 таблетки», «по 2 флакона раз в месяц», «внутривенно 40 ампул в день». Как бы странно на первый взгляд это ни выглядело, в некоторых обстоятельствах они действительно требуются, а значит, их надо иметь в справочнике.

Справочник операций и процедур. Когда требуется пополнить этот справочник, разработчику, прежде всего, необходимо полное наименование без сокращений и аббревиатур, привычных только специалистам. Например, не годится «ГУР» - нужно всем медикам понятная «трансуретральная резекция». Далее важно указать анатомическую область, в которой производится операции. Это нужно, чтобы облегчить врачу её поиск в справочнике.

Надо знать, процедура это или операция, а в последнем случае выделять «малые операции», то есть те, которые в истории болезни регистрируются и для учёта работы хирургов необходимы, но в официальный отчёт о хирургической деятельности не включаются. Наконец, поскольку в официальном отчёте необходимо выделять операции, относящиеся к высоким технологиям, да ещё по видам этих технологий (криохирургия, микрохирургия, стентирование, операции с видеоподдержкой и др.), в запросе надо указывать и эти особенности. Впрочем, поскольку и понятие «малых операций», и зачисление в «высокие технологии» подвержены нередким изменениям и даже могут зависеть от местных обстоятельств, разработчик может предоставить самим пользователям отражать в справочнике эти характеристики – достаточно разрешить медицинскому статистику доступ к соответствующим полям справочника.

Справочник лабораторных и других специальных методов обследования. Требования к дополнениям в этот справочник наиболее сложны. Помимо наименования исследования, в котором не должно быть сокращений и аббревиатур, кроме общеупотребительных (ЭКГ, МРТ, УЗИ и т.п.), здесь надо указывать группу, к которой относится исследование (клинические анализы, биохимические, серологические, иммунологические, эндоскопические, рентгенологические исследования и т.п.) Это необходимо для удобного поиска анализа в справочнике, для различных вариантов обработки массива историй болезни: для передачи назначения врача в лабораторию или кабинет, для отчётов о работе лабораторий, для анализа лабораторного обеспечения лечебно-диагностического процесса.

Мало указать наименование исследования – надо сообщить, как следует описывать его результаты в истории болезни и в лабораторном журнале. Пользователь должен задать формат описания, который будет предлагаться для ввода данных. В самом простом случае он состоит из поля для текстового протокола и поля для заключения. Для некоторых исследований надо обязательно указывать их номер по журналам лаборатории (таковы, например, анализы на реакцию Вассермана или ВИЧ). Есть исследования, результат которых выражается простым указанием на наличие или отсутствие определённого факта: «обнаружено – не обнаружено», «положительный результат – отрицательный результат». Результаты многих серологических анализов оцениваются одним или несколькими «кре-

стами» или таблицей с перечнем титров, где указывается, с какого титра лабораторная реакция становится положительной. Эти условия надо задать, продемонстрировать.

Для анализа, целью которого является один-единственный числовой результат, надо указать его максимально возможное значение (число необходимых символов), для символьного результата – все его варианты и в обоих случаях надо указать единицы измерения. Наиболее сложен формат многокомпонентных анализов: они могут включать как числовые, так и символьные компоненты. Для каждого компонента надо указать максимально возможное число символов и единицу измерения. К этому может добавляться символьная строка для дополнений и комментариев – надо указать и её максимальный размер. Разумеется, любой формат должен содержать поля для даты осуществления и кода лаборанта, но именно потому, что они нужны всегда, напоминать о них разработчику нет особой необходимости.

Чтобы ничего не было упущено, для запроса на лабораторное исследование следует использовать специальный макет, в котором остаётся заполнить подготовленные поля: для наименования анализа, для группы, к которой он относится, для наименования и единиц измерения каждого компонента, для числа необходимых символов.

1.3.5. Совершенствование функций, новые функции, изменения в структурах.

Сегодня самая слабая сторона информатизации здравоохранения – отсутствие детально проработанных универсальных представлений о том, что именно должна делать МИС, в каких функциях медицинских работников участвовать, каков полный набор требований, которые надо перед нею ставить. Негласно (а иногда и в голос) признаётся даже, будто универсального набора и быть не может, будто для каждого ЛПУ он таков, каким захотят его видеть работники данного учреждения. Из-за этого в полном противоречии с тем, что медицина и организация лечебного дела для всей России одна и та же, каждый разработчик руководствуется личными представлениями о должном. Потом уж суровая жизнь вынуждает вносить в эти представления то одни, то другие поправки, неуклонно подвигая разные разработки к некоему общему знаменателю.

Неудивительно, что запросы этой группы наиболее разнородны и наименее поддаются формализации: стартовые условия для эксплуатации МИС различны, а возникающие потребности целиком зависят от субъективных устремлений и активности отдельных пользователей. Но есть и категория запросов, которая определяется не этим, а общими для всех внешними обстоятельствами. Например, с середины 90-х годов прошлого века появились запросы на АРМы врачей общего профиля («семейных врачей»), после 2000-ного года потребовался автоматизированный учёт стоимости лечения и составление реестров

по ОМС и ДМС, затем появились льготные рецепты, талоны дополнительной информации в поликлинике, усложнились требования к вакцинациям в соответствии с Российским национальным календарём прививок, ужесточился учёт ВИЧ-инфицированных, потребовались непереносимые «информированные согласия» пациентов, - перечень можно продолжить. И, конечно, с постоянством, достойным лучшего применения, менялись официальные статистические отчёты ЛПУ. Для всего этого требовались новые функции МИС. И всё это в той или иной степени будет продолжаться.

Понять, что требуется сделать, когда речь идёт о совершенствовании функций или о совсем новых функциях МИС, намного сложнее, чем при тех видах запросов, которые мы уже рассмотрели, где хватало правильного оформления требований к разработчику на основе чётких правил и под руководством системного администратора. Для чего и как затрагивать функции МИС – это вопросы, решение которых часто требует личного общения разработчика с ответственным медиком, ставящим задачу.

Дело в том, что потребность всегда возникает под воздействием частной ситуации, предельно конкретной, но вызванные ею пожелания или требования могут иметь более общее значение, оказаться полезными и в других обстоятельствах. Описать возникшую конкретную потребность пользователь всегда способен, но с обобщениями дело обстоит иначе. Для них надо хорошо представлять устройство информационной системы и, сверх того, иметь склонность к системному взгляду на вещи и события, что гораздо больше свойственно тем, кто сам конструирует сложные системы, а не их пользователям.

Мы имеем здесь ситуацию, когда ни одна из сторон взаимодействия не обладает всей совокупностью знаний и умений, не может в одиночку представить, на какие области целесообразно экстраполировать новое решение, чтобы не возвращаться к нему потом в сходных ситуациях, когда один партнёр без другого не готов предусмотреть, чтобы изменения в одном звене системы не отозвались неблагоприятно в других звеньях. Тут необходимо понимание задачи и её решения вырабатывается только при личном контакте сторон, при очных встречах.

Степень сложности новых задач, затрагивающих функции МИС, определяется ответами на два вопроса. Первый: требуется совсем новая функция или развитие уже имеющейся? Второй: понадобятся ли структурные изменения в электронной истории болезни и в справочниках? Ответы не всегда лежат на поверхности. В большинстве случаев разработчик может это понять лишь после такого осмысления задачи, которое требует упомянутых выше очных встреч с медиками.

Развитие уже имеющейся функции и при этом не требующее изменений в структурных элементах МИС – обычно вполне ясная задача. Чаще всего она сводится к тому,

чтобы дополнить некоторыми данными ту или иную печатную или экранную форму: в списке пациентов требуется новая графа (например, для сведений о принадлежности к льготной категории или о тяжести), в официальном отчёте – дополнительные строки (например, для отдельной группы диагнозов или операций), на экране при работе с историей болезни - напоминания о давности болезни, сроке беременности, непереносимости медикамента и т.п. Всё это недвусмысленно, не чревато какими-либо побочными эффектами и требует сравнительно небольших усилий от программиста.

Запросы на совершенно новые функции разнородны. Когда дело сводится к созданию новой официальной отчётной формы, достаточно предъявить эту форму и инструкцию по её заполнению. Разработчику остаётся убедиться, что в базе данных МИС есть вся информация для этого отчёта, и можно приступать к работе. Если всей информации нет, задача серьёзно усложняется: приходится вносить добавления в структуру базы данных, а значит, создавать средства для ввода и просмотра соответствующих новых сведений.

Сложнее обстоит дело с функциями, которые понадобились самим врачам и руководителям для анализа и оценки собственной работы. Разумеется, в основу запроса кладутся изложенные письменно формы таблиц и списков, логические выкладки, предложения о месте, откуда надо будет вызывать функцию, но разработчик не может воспринимать всё это пассивно или ограничиваться только уточнением неясных деталей. В отличие от официальных отчётов, здесь важно понять, для каких целей нужны эти новые документы, как ими будут пользоваться, какие выводы будут делать из полученных данных.

Врачи, как правило, ставят задачу предельно узко и скромно. Разработчик должен компенсировать это своим умением мыслить системно, своими знаниями об устройстве МИС, об уже имеющихся в ней сходных по форме функциях, своим искусством программирования. Пользователям надо настойчиво внушать, что программист «может всё», лишь бы это «всё» было хорошо сформулировано.

Когда предстоит сделать новый список пациентов, надо позаботиться, чтобы каждая строка с фамилией сразу содержала всю полезную для данной функции информацию, а сам список был упорядочен наиболее подходящим для использования образом. При формировании статистических таблиц надо понять, что полагается считать по отдельным врачам, что следует суммировать в формах для подразделения и для учреждения в целом, какие данные будут использоваться для сравнительных оценок во времени и пространстве. Разработчик должен вникнуть в интерес пользователя активно, постараться развить его вширь и вглубь, а пользователь – лишний раз увидеть, на что способно программирование.

Как уже замечено, новшества могут потребовать изменений в структуре базы данных: добавления новых полей в электронной медицинской карте (истории болезни) и в справочниках или изменения формата уже имеющихся полей. Проиллюстрируем примерами связанные с этим последствия для разработчика.

До определённого времени не существовало понятия СНИЛС, но вот оно появилось и теперь СНИЛС пациента должен быть указан в каждой истории болезни. Значит, надо сделать для этого реквизита специальное поле, надо внести изменения в функцию ввода паспортных данных, надо проверить, в каких выходных формах этот реквизит выводить и, стало быть, внести поправки в соответствующие алгоритмы. Более того, на федеральном уровне предусмотрены правила кодирования СНИЛС и проверки введенных данных, которые также необходимо уточнить и реализовать в виде программной функции. Пользователю всё это без разработчика и в голову не придёт, а разработчик без пользователя может что-то упустить или наделать лишнего.

Более сложный пример - справочник госпитальных отделений. То, что для множества функций контроля, анализа, формирования отчётов надо характеризовать каждое отделение профилем (профилями) его коек и их штатным количеством, понятно с самого начала. Для этого в справочнике отводятся поля, и до конца прошлого века этого хватало. Но появились новые запросы у главных специалистов служб городского здравоохранения, у департаментов, появилось медицинское страхование со своими особыми требованиями. Эффективное управление коечным фондом стало особенно актуальной задачей, требующей разносторонней обработки данных. И тогда местный справочник пришлось дополнить полями для профиля специализированной службы, для вида финансирования, для плановой функции койки в году и планового среднего койко-дня, для отметок о профилях консультаций, которые способно оказывать данное отделение, для тарифов. Всё это надо было задействовать не только для отчёта раз в году, но и для ежемесячного слежения, для регулярных прогнозов, для сравнения отделений между собою, для оперативного управления использованием коечного фонда со стороны заведующих отделениями и главного врача, для взаимодействия отделений с поликлиникой.

Эти примеры показывают, что если разработчик в некоторой мере вник в то, как устроена и функционирует больница, то с его системным мышлением вполне можно ставить дополнительные вопросы о предстоящем вмешательстве в функции МИС и делать некоторые свои предложения, имеющие целью как более точное и объёмное решение медицинской задачи, так и наиболее адекватное программистское решение.

Отдельно следует сказать о проблемах, связанных с правкой кода данных и другими вмешательствами в структуру базы данных. Когда архитектура МИС заранее хорошо

продумана и многое сделано с запасом, на вырост, эти крайне нежелательные изменения требуются редко, но застраховаться от них полностью невозможно. Для каждого поля в структуре базы данных устанавливается, какой тип данных можно в него вводить, а также ограничивается максимальный размер данных. Этим достигается рациональное использование ресурсов и в значительной степени обеспечивается правильность ввода данных, заведомое исключение множества ошибок при вводе. На избранный вариант настраиваются программы, работающие с этими данными. Изменение типа или размера поля означают необходимость поправок во всех таких программах. Приходится пересматривать все соответствующие функции, чтобы предупредить возникновение несоответствий во время работы пользователя.

Так, всем, кто свои программы сделал ещё при Международной классификации болезней 9-го пересмотра, пришлось при переходе к МКБ-10 не только полностью заменить справочник диагнозов и сформировать новые официальные отчёты, но и переделать функцию ввода диагноза в историю болезни, все документы, куда включается диагноз, все функции выборки из историй болезни по диагнозам. Другой пример: одному из авторов этих строк пришлось столкнуться с ситуацией, когда в госпитальной истории болезни появилось, казалось бы, немыслимое количество общих анализов крови – более 99, а поле для нумерации этих строк было рассчитано только на 2 символа. За 20 лет эксплуатации системы такая ситуация возникла впервые, но если уж она могла возникнуть раз, то непременно когда-нибудь и повторится. Пришлось изменить не только формат электронной истории болезни, но вмешаться и в программный код, и в структуру вторичных по отношению к истории болезни выходных форм.

Самое же неприятное при изменениях этого рода состоит в том, что преобразованная в соответствии с новым изменением программа должна правильно работать и с накопленным до изменений архивом историй болезни, а там формат старый. Это может приводить к фатальным коллизиям. В редких случаях их удаётся решить с помощью программных ухищрений, но чаще приходится прибегать к радикальному средству – одномоментно переформатировать записи в базе данных, приводя их к новому формату хранения. В любом случае, от разработчика требуются и тщательное обдумывание, и трудоёмкие действия, и надежные испытания и тестирование.

Наконец, стоит иметь в виду, что последствия вмешательства в действующую структуру могут быть очень многообразны и разнородны. Кое-что поначалу может остаться вне поля зрения и проявится только впоследствии, внезапно, в виде неожиданной ошибки. Поэтому изменения в структуре особенно важно запоминать, чтобы при таких отдалённых последствиях быстро распознать их природу.

1.3.6. Замечания о дизайне и интерфейсе

Система общается с пользователем посредством соответствующего интерфейса: экрана и распечаток. Как выглядит экран или печатный бланк, насколько он информативен в каждый момент работы, не перегружен ли, не содержит ли лишнего, понятна ли, не двусмысленна ли выдаваемая им информация, - всё это очень важно, так как речь идёт об инструменте, с помощью которого осуществляется интенсивная повседневная работа. Но все ответы на такие вопросы окрашены субъективным отношением, личными пристрастиями, а о вкусах не спорят и это правило надо помнить, когда пользователь предлагает изменить вид экрана, распечатку или способы работы за компьютером.

Если одним нравится так, а другим – иначе, надо понять, что за этим стоит: индивидуальные привычки или объективные неудобства, с которыми одни не мирятся, а другие решили потерпеть. Привычки – дело наживное, объективные же неудобства всегда будут снижать работоспособность системы и её пользователя. С этой точки зрения и надо оценивать претензии к дизайну: к шрифтам, цветовой гамме, пестроте, перегруженности экрана. Знакомясь с ними, надо всякий раз убедиться, что у других пользователей предлагаемые изменения не вызовут противоположных замечаний. Намеченные перемены надо показать не только самим авторам, но и их коллегам. В спорных случаях всегда лучше оставить тот вариант, который соответствует предпочтениям самого разработчика. Это означает, что далеко не все предложения по изменению дизайна будут реализованы разработчиком, к этому пользователи должны быть готовы.

В отношении претензий к способам взаимодействия пользователя с МИС можно использовать более определённые критерии. Всё, что обещает повысить эргономичность работы врача, надо использовать. Там, где можно уменьшить число нажатий клавиш и щелчков мышью, надо его уменьшить. Там, где для определённого действия одному удобно щёлкнуть мышкой, а другому привычнее нажать функциональную или «горячую» клавишу, надо уклониться от споров, сделав доступными обе возможности.

Перечни, из которых надо делать выбор, должны упорядочиваться так, чтобы обеспечить удобства обзора и выбора, в ряде случаев надо дать пользователю возможность упорядочивания то по одному признаку, то по другому, особенно обширные справочники следует снабжать специальными средствами поиска. Надо широко использовать поиск в справочниках по ключевым словам.

Пользователь обычно не ставит задачи именно так, он просто жалуется на явное неудобство. Разработчик, посмотрев и расспросив, должен сам классифицировать проблему и подобрать к ней наилучшее профессиональное решение.

Всяческого внимания заслуживают претензии к инструкциям, пояснениям и подсказкам на экране. Если тех, для кого они выглядят двусмысленными, не вполне определёнными, меньшинство, это ничего не значит. Инструкции должны быть понятны с первого взгляда и всем, кто так или иначе работает в системе вне зависимости от их специального образования: врачам, медицинским сёстрам, медицинским статистикам, администратору МИС.

У экранных форм есть совершенно специфическая функция – напоминать врачам во время работы о том, что заслуживает внимания именно сейчас: о пропущенных сроках, об отсутствии обязательных сведений, о факторах риска у пациента, о том, какие ловушки важно не проглядеть при постановке некоторых диагнозов. Задачи этого рода ставят только очень немногие, только особенно требовательные врачи и лишь тогда, когда к ним приходит понимание настоящих возможностей программиста. Вот этому пониманию надо всячески содействовать. С другой стороны, и разработчику чрезвычайно полезно знать о таких конкретных упущениях, случающихся в практике врачей, которые связаны с памятью. Забыть может всякий, но компьютер напомнит всегда о том, чему его научили. Таким образом, программист может оказать врачам ценные услуги, если ему известны ситуации, в которых забывчивость проявляется часто или где она может привести к тяжёлым последствиям. Владея такой информацией (а её можно получить, если интересоваться ошибками, которые врачи разбирают в своих коллективах), разработчик способен усилить положительное воздействие МИС на качество медицинской помощи.

Понятно, что в отношении дизайна и интерфейса справедливо то же, что было сказано о дополнении и совершенствовании функций: для полного понимания запросов и способов их удовлетворения здесь часто требуется прибегать к личным, очным контактам разработчика и пользователя. В личном общении оба детальнее понимают конкретную проблему и возможности её решения и, сверх того, вообще глубже вникают в задачи и возможности автоматизации лечебно-диагностического процесса.

1.4. Основания для отклонения запроса.

Не всё, о чём попросят, надо делать. Есть несколько групп ситуаций, когда запрос следует обоснованно отклонить.

Нельзя реагировать на устные замечания, которые делают на ходу, - время для них истекло с окончанием этапа ввода МИС. В большинстве случаев не следует принимать и

критику тех, кто лишь наблюдает, а сам в системе не работает: спровоцированные ими изменения их не затронут, поэтому их предложениям часто недостаёт ответственности.

Как правило, надо отклонять предложения что-то удалить из системы только потому, что это нечто представляется лишним тому или иному работнику. За такими просьбами почти всегда стоят либо старые привычки, либо отсутствие системного взгляда на ситуацию. Лишним могут посчитать то, что не используется в данном ЛПУ, но нужно другим, либо просто редко встречается, но именно поэтому оно и должно быть наготове. То, что врачу не требуется для работы в своей узкой специальности, может понадобиться и пригодиться при вынужденном выходе в смежные области. То, что не нужно лечащему врачу, полезно руководителю. В конце концов, просто не надо пользоваться тем, что вам не нужно, - вот самый частый ответ в таких случаях.

Те запросы, для оформления которых установлены правила и формы, при нарушении этих правил надо отклонять, не делая исключений. Иначе привычка соблюдать правила не выработается никогда. Однако нарушение правил – это повод не «отфутболить» активного человека, а побудить его, помочь ему всё переделать по правилам и непременно донести свою мысль до разработчика. Активность пользователей нельзя подавлять формализмом, наоборот, ей надо содействовать, но направлять в верное русло.

Не следует руководствоваться только вкусами отдельных энергичных людей: за ними часто стоят всего лишь старые привычки, утратившие значение в условиях автоматизации. Особенно это касается форм медицинских документов и шаблонов. Часто просят, например, заканчивать описание статуса пациента диагнозом и планом лечения, хотя это бессмысленно и даже вредно, потому что и диагнозы, и назначения врача вводятся в специальные места истории болезни, откуда и выводятся потом в экранные и печатные формы.

Как правило, предложения об изменениях форм документов и текстовых шаблонов выдвигаются лидерами врачебных коллективов, формальными или неформальными, но это ещё не достаточное основание, чтобы с ними соглашаться, тем более, что как раз лидеры особенно отстаивают именно привычки. То, что уже имеется в системе, попало туда тоже по инициативе таких же лидеров. Уважение к их труду не позволяет отменять или изменять ими сделанное без их согласия, если они в пределах досягаемости, а если нет, то надо либо убедиться, что новые предложения явно обогащают уже имеющееся содержание или совершенствуют его форму, либо их отклонить. Разработчик должен подчеркивать своё бережное отношение к уже сделанному вкладу врачей, давая тем самым понять, что и новый полезный вклад будет охраняться так же.

В большинстве случаев разработчик сам может оценить содержательность добавлений и поправок в документы и тексты, но при необходимости надо прибегать к третьей-стороннему суду. Решающим тут является голос такого специалиста, который одновременно является руководителем, то есть несёт ответственность за качество лечебно-диагностического процесса, включая культуру документации. Круг таких лиц ограничен: заведующий отделением, заместитель главного врача по данному разделу медицины, сам главный врач, главный специалист города, руководитель клинической кафедры на базе данного ЛПУ. Не стоит смущаться затратами на встречи с этими людьми – совсем не грех дать им понять серьёзность заботы разработчика о поддержке качества МИС.

По здравому смыслу, запрос пользователя надо безусловно удовлетворять, если он представляет собою вклад в медицинское содержание МИС, в культуру медицинских записей, в удобство работы, в управление деятельностью больницы, если он теми или иными из этих путей участвует в совершенствовании медицинской помощи. Казалось бы, верно и обратное: если ничего перечисленного в запросе нет, его надо отклонить. На деле не так. На деле существуют ситуации, когда приходится вносить в МИС и то, что заведомо не идет на пользу ни врачам, ни их руководителям, ни государственной статистике.

Такие ситуации - результат неравномерности информатизации здравоохранения. Ушедшие вперёд ЛПУ уже решают информационные задачи без старых средств, а органы здравоохранения требуют предъявлять им эти средства по привычке. Когда не удаётся доказать им, что в условиях автоматизации способы контроля надо менять, освобождая сотрудников от старых бумаг, ничего не остаётся, как, вопреки логике, формировать ненужные больнице талоны амбулаторного пациента, карты, журналы учета, - всё только затем, чтобы медикам не пришлось делать это вручную, дабы избежать нареканий «свыше» и даже штрафных санкций. Но помнить, что это – лишь вынужденный выход из обстоятельств, которые не будут вечными, что с течением времени такие функции станут балластом, бесполезно. Поэтому определим их, разделив на две основные группы.

Одна – это формирование официальных учётных форм. Смысл последних только в том, чтобы с их помощью составлять отчёты: бумажная история болезни для этого непригодна. Но история электронная как раз и делается так, чтобы из введенных в неё данных можно было создавать любые отчёты. Она по своей сути – универсальная учётная форма, позволяющая автоматизировать всю официальную отчётность, при этом и сама она легко поддаётся контролю. Значит, напечатанные с помощью МИС учётные формы ничего нового никому не дают.

Другая группа запрашиваемых функций, не нужных лечебному учреждению, - это так называемые «мониторинги». С их помощью органы здравоохранения или медицин-

ского страхования хотят следить за правильностью текущей работы врачей. Здесь порочна основная идея - за работой врачей следят, управляют ею другие фигуры: заведующие отделениями, главный врач, главные специалисты. Это не только их компетенция – это их ответственность. Они, в отличие от потребителей «мониторингов», хорошо знают, за чем надо следить и как на увиденное реагировать. МИС обеспечивает их всей необходимой для контроля информацией, всегда неизмеримо более богатой и глубже проработанной, чем то, что сегодня есть в «мониторингах». Кстати, те, кто запрашивает «мониторинги, могли бы для своих целей воспользоваться именно этой готовой информацией вместо того, чтобы нагружать больницу дополнительной работой, ничего ей самой не дающей.

Можно, конечно, сказать, что у органов здравоохранения и плательщиков возникли сложности: в сфере их ответственности одни учреждения уже используют МИС, а другие нет. Выход очевиден - дифференцированное отношение к тем и другим. Но это – новая и специальная задача, вникать в неё то ли не хотят, то ли некому, - судить не нам. Ясно, что решения ее сейчас не будет. Так что пока информатизация здравоохранения не станет всеобщей, пионеры будут вынуждены тратить силы на то, что лишено смысла. Со временем это пройдет, а встроенные в МИС «мониторинги» станут мусором, информационным шумом, от которого можно будет освободиться.

1.5. Передача запросов.

Между заявлением о потребности у пользователя и временем, когда запрос в окончательной форме принимается разработчиком, лежит период, в течение которого надо обоюдными усилиями сторон обеспечить взаимопонимание, взаимопроникновение. Здесь необходимо, чтобы с каждой стороны действовал ответственный, постоянный и коммуникабельный сотрудник. Задача обоих – следить за правильным оформлением и обоснованностью запросов, но делать это так, чтобы никоим образом не гасить активность пользователей. Наоборот, оба должны терпеливо и доброжелательно помогать им выразить свои желания наиболее полно и точно. На стороне ЛПУ важно, ухватившись за однажды высказанное недовольство, выявить его конкретную причину, на стороне разработчика – осмыслить значение предполагаемых перемен во всей системе и затем принять решение: либо поставить конкретные задачи программисту, либо мотивированно отклонить предложение, либо предложить иной способ решения проблемы. Потом, когда свою работу сделает программист, оба должны – каждый на своей стороне – убедиться в приемлемости и доброкачественности результата.

В ЛПУ всё это логично поручить системному администратору, а если есть заместитель главного врача по информационному обеспечению, которому подчинены системный

администратор и медицинские статистики, то, конечно, лично ему. В компании-разработчике это должен быть руководитель, отвечающий за взаимодействие компании с заказчиком и пользователями и при этом имеющий влияние на процесс разработки системы. Как правило, это руководитель службы Service Desk.

Взаимодействие должно обеспечиваться постоянной связью, позволяющей передавать письменные запросы, а также файлы с необходимыми материалами. Электронная почта подходит для этого как нельзя лучше. Но важна и личная связь, позволяющая обсуждать проблему устно, а при необходимости - демонстрировать её и способы её решения на экране. Когда нельзя это сделать непосредственно, сегодня уже можно прибегнуть к техническим средствам, позволяющим не только видеть друг друга на дистанции, но и работать на удалённом компьютере.

2. УДОВЛЕТВОРЕНИЕ ЗАПРОСОВ ПОЛЬЗОВАТЕЛЕЙ.

Когда запрос сделан, проверен на правильность и полноту, при необходимости обсуждён и, наконец, принят разработчиком как руководство к действию, наступает черёд программиста: он вносит в систему изменения и тестирует сделанное. Далее, всякое изменение должно быть кратко зарегистрировано в документе «Список выполненных изменений», чтобы администратор мог видеть всё новое, чем отличается очередной релиз МИС. Кроме того, если изменение состоит в добавлении или совершенствовании функций, его необходимо отразить в «Руководстве пользователя».

Затем надо передать обновлённый вариант МИС туда, откуда поступил запрос, оповестить об этом системного администратора и вскоре получить от него подтверждение, что авторы запроса удовлетворены. Иногда вместо удовлетворения приходят рекламации, указания на недоработки или нежелательные побочные эффекты – тогда цикл повторяется. Наконец, если пользователи довольны, обновлённая МИС устанавливается во всех учреждениях, где она используется, а учреждения оповещаются об обновлении.

Когда разработчик, переговорщик, постановщик задачи и программист – это один-единственный человек (а авторы по личному опыту знают, что так бывало), для быстрой и безотказной работы описанной схемы требовалось немного – личная ответственность и самодисциплина, внутренняя организованность. Надо было просто понимать, что вас ждут врачи, и ничего не откладывать.

Положение существенно усложняется, когда речь идёт о компании-разработчике, где функции распределены между сотрудниками и даже разные поправки в систему вносят разные программисты. Лечебное учреждение всё равно должно знать только одного человека, конечная ответственность за удовлетворение запроса всё равно лежит именно на нём, но теперь он уже не может обойтись своей собственной организованностью. Он зависим от других сотрудников и для успеха нужна специальная организация работы: распределение обязанностей и прав, субординация, регламент.

2.1. Организация, инструменты, специалисты.

2.1.1. Общие условия.

Хотя все больницы и поликлиники выполняют одну и ту же работу и организованы одним и тем же образом, предназначенные для них МИС значительно различаются и по набору функций, и по своему устройству. Что касается функций, то под давлением требо-

ваний жизни они пополняются и будут стремиться к единому, общему для медицинской практики набору. Конструктивные же различия остаются неизменными: архитектура МИС, структура базы данных, система управления базой данных, язык программирования, алгоритмы программ, - всё это каждый разработчик выбирал из существующих у него возможностей по своему усмотрению. Добавим, что и компании-разработчики тоже функционируют каждая на свой лад и организация взаимодействия с пользователями тоже в каждом случае своя. Предложить здесь вариант, идеальный для всех, невозможно. Остаётся ограничиться изложением общих целей, которые должна преследовать организация, оптимальных установок и отдельных конкретных приёмов, а заодно рассмотреть некоторые спорные идеи.

Когда запрос пользователя понят, принят и готов к передаче программистам, надо руководствоваться стремлением сделать всё возможно скорее, без задержек, понимая, что результат необходим действующему производству, что оно уже споткнулось о проблему, которую просят решить, что, пока решения нет, возможны повторения этих спотыканий у всех, кто пользуется данной МИС. Отсюда мера ответственности.

Удовлетворение запроса почти никогда не означает работу с нуля. В части случаев всё сводится к редакционным поправкам: к устранению дефектов в текстах или в программном коде. Задача не сложная, надо лишь эти дефекты локализовать везде, где они могут быть, а не только там, где в этот раз на них натолкнулся пользователь. Но и сложную задачу, такую, как программирование новой функции, тоже часто удаётся решить не с нуля. Новый список пациентов во многих чертах похож на какие-то из уже имеющихся, новый отчёт содержит элементы, уже сделанные однажды для других отчётов, в новых умозаключениях используются те выборки из базы данных и те сопоставления, которые уже запрограммированы для других видов анализа. Преподобней, ранее сделанной работой надо воспользоваться – заимствовать алгоритмы, задействовать общие, уже давно отлаженные программные модули (функции, процедуры, подпрограммы).

Чем более развита МИС, чем больше уже сделано и чем лучше структурировано программное обеспечение, тем реже приходится программисту изобретать, делать что-то совсем новое, тем быстрее и надёжнее удовлетворяется новый запрос пользователя. Но это означает, что программисты, работающие над запросами, должны хорошо знать всю систему. Они должны уметь легко отыскивать аналогии тому, что запрошено, У них должно быть право на упомянутые заимствования, а иногда и на преобразования того, что сделано ранее (таковы, например создание или развитие функций и процедур, общих для старого и нового). С другой стороны, организация работ должна предусматривать, что при

подобных заимствованиях и преобразованиях есть риск нарушить старые функции и стараться предупреждать такую опасность.

2.1.2. Особенности дополнения системных справочников

Дополнение системных справочников требует вполне определённых действий. В общем виде это выглядит предельно просто: пользователь предлагает новое наименование, программист включает его в справочник и присваивает ему уникальный код, причём последнее обычно автоматически делает сама СУБД. Тем не менее, есть несколько моментов, которые надо оговорить.

В справочниках важно не допустить дублирования, и, как бы ни следили за этим другие при оформлении запроса, всегда полезно и программисту проверить, действительно ли в системе нет того, о чём просит пользователь. Для этого ему необходимы средства, позволяющие упорядочивать справочник по именам и искать в нём заданные наименования или их фрагменты. Кроме того, надо учитывать, что для многих медицинских наименований существуют синонимы, а также сокращённые варианты и аббревиатуры. Нельзя, чтобы в справочнике оказались одновременно «Электрокардиография» и «ЭКГ», «Офтальмолог» и «Окулист», «Ацетилсалициловая кислота» и «Аспирин» и т.д.

Не всегда просто и с кодами. Не во всех справочниках можно полагаться только на саму СУБД. Пример – коды диагнозов по МКБ-10, в последовательности которых допускаются пропуски, а сама последовательность опирается не только на естественный числовой ряд, но и на содержательные основания. Для таких справочников надо либо формировать уникальный код программно, либо, если код задаётся самим пользователем (как в случае с добавлениями в МКБ), иметь средство проверки его уникальности.

Дополнять справочники, в которых, кроме наименования и кода, содержатся поля для другой информация, надо не вручную, а с помощью специальных программ, требующих заполнения этих полей и помогающих их правильно заполнить. В части таких случаев для ввода сведений в эти поля нужны вспомогательные перечни, которые предлагаются на экране и позволяют правильно указать для операции анатомическую область, для осложнения – клиническую сферу, для медикамента – клинико-фармакологическую группу, лекарственную форму, дозу, возрастные ограничения, комментарии.

2.1.3. О соотношении между системными справочниками и официальными классификаторами.

С появлением официальных классификаторов («Федеральная нормативно-справочная информация»), отдельные классификаторы, предлагаемые департаментами

здравоохранения и территориальными ФОМС) возникла идея их-то и включать в МИС в качестве её справочников. Казалось бы, так можно разом снять с МИС все заботы и о первоначальном составлении справочника и о его поддержке. Но это заманчиво только на первый и поверхностный взгляд.

Прежде всего, примем во внимание, что такие классификаторы, даже если их утверждают серьёзные инстанции, ещё далеко не обкатаны, а потому подвержены внезапным и серьёзным изменениям. Ещё существеннее, что справочники системы и официальные классификаторы создаются для разных целей и адресованы разным деятелям. Всеобъемлющие классификаторы нужны органам здравоохранения и государственной статистики для учёта, планирования и анализа затрат и расходов в масштабе страны, но отнюдь не для управления работой лечебных учреждений. У справочников МИС другие задачи: они, прежде всего, инструмент для ввода информации в историю болезни, в них должно быть только то, что достаточно для работы врачей.

С точки зрения нужд каждого лечебного учреждения, официальные классификаторы неоправданно громоздки. Огромные излишества затрудняют работу врачей и при этом не страхуют ни от неудачных, не рассчитанных на работу врача принципов классификации, ни от ошибок в наименованиях, ни от дефектных наименований, ни даже от неполноты сведений. Между тем, внести сюда необходимые врачам исправления никто не может. В противоположность этому, собственные справочники МИС составляются и своевременно пополняются в соответствии с потребностями лечебных учреждений.

С другой стороны, в некоторых случаях к ЛПУ предъявляются требования отчитаться именно в терминах официальных классификаторов. Выходом из такого положения, когда требуется перевод с одних кодов на другие, служит то, что нужно вообще для перевода с одного языка на другой, – двуязычный словарь. Проще всего это достигается тем, что в справочнике МИС предусматривают отдельную графу для кодов официального классификатора. По мере надобности эту графу заполняют медицинские статистики.

2.1.4. О конструкторах форм.

Нередко компании-разработчики создают «конструкторы форм» - специальные программы, которые позволяют разрабатывать новые экранные и печатные формы, опираясь на структуру базы данных, в основном, - на структуру электронной медицинской карты (истории болезни). Для форм, которые не требуют изменять эту структуру, такое решение очень полезно, оно уменьшает сроки разработки, избавляет от ряда ошибок, экономит затраты разработчика.

Однако с появлением хорошего конструктора возникает соблазн включить его в состав МИС и передать пользователям, чтобы они сами создавали себе всё, что им нужно: новые списки, аналитические таблицы и официальные отчёты. Казалось бы, это разумно: и разработчик освободится, и у пользователя руки развязаны, ЛПУ становится менее зависимым от компании-разработчика. Даже если средств на сопровождения у ЛПУ не окажется, оно всё-таки сможет что-то дорабатывать в своей системе.

Тем не менее, такое применение принципа «спасение утопающих - дело рук самих утопающих» заслуживает оговорок и ограничений. Прежде всего, разработчик МИС должен не просто предоставить рабочий инструмент, но и позаботиться об ответственности за его использование. Каким бы опытным ни был администратор МИС в ЛПУ, он всё равно не обладает тем уровнем знаний архитектуры МИС и опытом, которые позволяли бы ему обеспечить не только правильность работы новых форм, но и отсутствие скрытого вреда для всей системы, тем более, что запросы могут быть весьма необычными: предвидеть все повороты мысли тех, кто предлагает новые формы, не в состоянии никто. Поэтому создание универсального конструктора потребует от разработчиков предусмотреть самые разнообразные нештатные ситуации и обезопасить систему от их возможного нежелательного влияния.

Но пусть конструктор, настолько хороший, что его можно передать в ЛПУ, создан. Теперь возникает сомнение в правомерности такой передачи обязанностей разработчика пользователям. Для последних ясно, что создание новых форм – не просто чужая, а чуждая им работа. Она никак не относится к их профессии, у них нет необходимых знаний и навыков и, к слову, эти навыки никогда не выработаются, потому что использовать конструктор форм будут не систематически, а редко, от случая к случаю. Ответственность же за результат велика. Всё это, очевидно, создаёт почву для отторжения. Дело может обернуться так, что там, где новая отчётная форма, новый список или аналитическая таблица потребуются позарез, их будут составлять вручную, на бумаге, врачи или статистики, а там, где можно, обойдутся без них.

Частично это настроение можно ослабить, строго установив, что работа с конструктором поручается только ИТ-работникам. Но если под ИТ-работниками понимать не системного администратора и не медицинских статистиков, а других специалистов, имеющих особую подготовку, то кто они, кто ими руководит и вообще есть ли они в штате больницы или поликлиники? Если есть, то было бы логично подчинить их функционально разработчику (подобно тому, как статистики ЛПУ подчинены статотделу органа здравоохранения).

Наконец, принципиальным недостатком передачи конструктора форм в ЛПУ является то, что сделанные одним учреждением доработки недоступны другим, которые со временем будут изобретать велосипед. Мало того, что это нерациональная трата сил. Здесь сводится на нет уникальный эффект взаимодействия пользователей с разработчиком, заключающегося в том, что все исправления и полезные дополнения, сделанные с подачи одного ЛПУ, сразу становится достоянием всех других учреждений, использующих ту же систему. Это относится и к полноте справочников, и к культуре текстов, и к новым функциям, включая составлению форм, среди которых, помимо официальных отчётов, есть найденные самими медиками отдельного учреждения удачные решения, до которых другие не додумались, но которые этим другим смогут пригодиться тоже. Это специфическое и ценнейшее свойство МИС поднимать, подтягивать всех к тому более высокому уровню, куда сначала шагнул кто-то один, надо использовать во всей сфере влияния компании-разработчика. Передача функции создания новых форм самим ЛПУ этому противостоит. Если мы не хотим такого противостояния, надо организовать дело так, чтобы всё, что сделано в ЛПУ и оказалось полезным, поступало к разработчику, а от него – во все другие учреждения.

Таким образом, если идею конструктора форм для ЛПУ реализовывать всерьёз, то, во-первых, её не надо понимать как способ избавиться от определённой группы контактов, а, во-вторых, она должна вызвать к жизни ряд организационных решений, затрагивающих права и обязанности сотрудников обеих сторон. В противном случае желаемого результата можно не достичь, а вредные побочные эффекты будут.

2.1.5. О вводе текстовых шаблонов.

У текстовых шаблонов, предназначенных для ввода в историю болезни, несколько задач. Они должны значительно облегчать работу врача за счёт пресловутой «писанины». Они обеспечивают общую культуру текста и его медицинскую полноту. А ещё они представляют собой программу действий, напоминают, что именно должен врач выяснить о больном, на что обратить внимание, что и как описать. Первоначально предложенные врачами тексты не всегда соответствуют этим задачам. Они нуждаются в литературной, медицинской и технической редакции.

Литературная редакция сводится к орфографии, синтаксису и устранению сокращений, не являющихся общепринятыми. Под редакцией медицинской имеется в виду забота о полноте описания ситуации, для которой предназначен шаблон, Что же касается технической редакции, то она преследует две задачи. Одна – сделать так, чтобы превращение шаблона в описание реальной ситуации требовало от врача минимума действий.

Предложения в тексте строятся так, чтобы избежать оборотов, где требуются родовые окончания, чтобы не приспособливаться к полу пациента. Для характеристики того или иного состояния предлагаются все возможные альтернативы с таким расчётом, чтобы ненужные было легко стереть, не нарушая конструкции фразы. Из двух альтернатив на первое место ставят наиболее частую. Вообще, везде, где возможно, предпочитают такой вариант, при котором надо стереть лишнее, а не дописывать недостающее. Иногда бывает выгодно дать просто два варианта сложного текста, один из которых убирается целиком.

Другая задача технической редакции - позаботиться, чтобы врач действительно поработал во всех тех местах шаблона, где нужно внести реальные сведения. Эти места специально выделяются, желательно так, чтобы в случаях, когда к ним не прикоснулись, можно было сообщить об этом на экране и потребовать действий. Без подобных приёмов нельзя исключить конфуза, когда в истории болезни обнаруживают не описание, а необработанный шаблон.

2.1.6. О консультантах.

Из всего изложенного в этом разделе видно, что для такого удовлетворения запросов пользователей, благодаря которому МИС обогащается полезными сведениями и функциями, совершенствуется, развивается вместе с медициной, нужны некоторые особые условия. Конечно, изменения в систему должны вносить опытные программисты. Но под опытностью здесь надо иметь в виду не только квалификацию вообще, но и хорошее знание всей МИС, её архитектуры, структуры базы данных, состава её медицинских функций, особенностей её функционирования в производственной эксплуатации.

То обстоятельство, что запросы пользователя – это не рассуждения об общих принципах, а всегда задача, возникшая из приложения информационной технологии к конкретной медицинской ситуации, задача на стыке программирования с медициной, означает, что для её наилучшего решения разработчику системы нередко нужны советы и заключения квалифицированных и авторитетных медиков. В поле зрения надо иметь соответствующих консультантов, причём таких, которые имеют достаточное представление о МИС (лучше всего, если они сами ею пользуются). Это надо предусматривать при организации всей работы с запросами пользователей.

2.2. Сроки удовлетворения запросов.

Сколько времени нужно на удовлетворение запроса, полученного от лечебного учреждения? Возьмём за точку отсчёта уже упоминавшуюся ситуацию, когда разработчик, программист, постановщик медицинских задач, – всё это один-единственный человек, постоянно находящийся на связи по электронной почте и взявший за правило приступать к работе над запросом немедленно после его получения. Предположим также, что технические свойства МИС позволяют тут же сделать инсталлятор с обновлениями, передать его по электронной почте системному администратору, после чего последнему достаточно для установки нескольких минут. Хотя это действительно возможно и даже сегодня есть некоторые ЛПУ, взаимодействующие с разработчиком именно так, подчеркнём, что в нашем рассуждении такая ситуация интересна не как пример для подражания, а лишь для первоначального уяснения задачи в её чистом, простейшем виде.

При заданных условиях затраты на исправление явных ошибок, опечаток, аварийных завершений, а также на дополнение справочников и шаблонов исчисляются минутами, от силы одним-двумя часами. Изменения в функциях – это затраты другого порядка. На их обдумывание, поиск уже имеющихся в МИС аналогов, реализацию требуются часы, иногда несколько дней. Несколько часов может потребоваться на тестирование. Подготовка инсталляторов, устанавливающих обновление на местах и передача их по электронной почте – это опять только минуты. Учтём далее, что передача, приём и инсталляция сделанного не всегда осуществляются немедленно. У системного администратора рабочий день ограничен, в выходные дни в больнице можно не всё, а ещё разработчик и пользователь могут находиться в разных часовых поясах. Наконец, для инсталляции обновления может понадобиться остановить работу системы, а для этого надо подобрать «окно», когда пользователей можно безболезненно от неё отключить; при этом приходится учитывать недельный ритм работы ЛПУ. Таким образом, то, что сделано за минуты, на пути к пользователям может задержаться, но в большинстве случаев удовлетворение запросов удаётся обеспечивать по принципу «к утру или к понедельнику». Именно так.

«Нереально!», - сказал себе уже несколько раз бывалый читатель. Да, когда разработчик – это компания, заказчиков – сотни, а удовлетворение запросов – труд коллективный, сроки слишком часто выглядят далеко не так.

Но почему собственно? На что ещё уходит время? В общем виде – на трение, на преодоление несогласованностей во времени, пространстве, на изучение и оценку возможностей реализации, на ожидание в очереди среди многих других изменений, требующих одних и тех же ресурсов, на контроль качества внутри компании. Однако для того и существует организация коллективной работы, чтобы обеспечивать согласованные дей-

ствия участников. Круглосуточный приём правильно оформленных запросов, взаимозаменяемость программистов, если каждый закрепляется за определённым этапом общего процесса, квалификация программистов, знание ими всех тонкостей МИС, чёткая ответственность за сроки и качество выполняемых операций, регламент для всего конвейера, по которому движется запрос от поступления до передачи результата в ЛПУ включительно, - всё это вопросы организации работ, не более того.

Иными словами, затраты времени сверх описанного выше в качестве точки отсчёта - это показатель уровня организации: чем меньше затраты, тем лучше организация. Получается, что правило «к утру или к понедельнику» не стоит относить к числу недостижимых фантазий.

А, между тем, забота о сроках имеет прямое отношение к требовательности пользователей, к их реакции на недостатки МИС в будущем. Удовлетворять запрос тогда, когда о нём уже забыли, а к дефекту притерпелись, - верный способ отбить охоту связываться с компанией в другой раз. Наоборот, неожиданно быстрая, демонстративно быстрая реакция удивляет пользователя и стимулирует его к дальнейшей активности. Тогда разработчик может при случае позволить себе что-нибудь вроде следующего диалога.

Врач. Ой, спасибо! Как быстро! За один день! А то мы мучились.

Разработчик. Пожалуйста, пожалуйста. А мучились давно?

Врач. Да уже месяца два, как это обнаружилось.

Разработчик. Прекрасно! Значит 59 дней ваших неудобств – за ваш счёт, а один день – за мой.

В связи со сказанным, надо сделать одну оговорку. Мы не учли время, которое уходит на уточнение запроса с его инициатором, с системным администратором, с консультантами. Здесь тоже нужна хорошая организация, включая уже упоминавшиеся правила оформления, причём она – забота обеих сторон. А главное - пользователь участвует в процессе уточнений, он видит, что сам не обеспечил сразу всю полноту информации и что дело движется. В таком случае затраченное время не ощущается как ожидание, не относится на счёт разработчика, не имеет отношения к его репутации.

К сожалению, сегодняшняя практика очень далека от описанной картины. Разработчик, особенно крупный разработчик, обслуживающий сотни лечебных учреждений, может получать такой объём запросов и предложений, что у него не хватает ресурсов, чтобы к удовлетворению каждого приступить немедленно. Возникает очередь, приходится расставлять приоритеты, сроки удовлетворения значительно возрастают и могут составлять многие дни и недели, а иногда, по признаниям самих разработчиков, даже месяцы. Неудивительно, что с пользователями возникают конфликты. Ещё тревожнее, что у них

пропадает желание проявлять активность, а разработчик в такой ситуации получает ложное представление, будто с эксплуатацией всё хорошо. Описанная выше нормальная схема явно извращается.

Первая причина такого положения – на поверхности. Разработчик не рассчитал своих возможностей, не учёл работу по сопровождению программ, когда определял масштабы внедрения своего продукта. В конце концов, выход сводится к тому, чтобы увеличить число штатных сотрудников, занятых в сопровождении МИС. Но увеличение это, по видимому, не маленькое, в разы, а потому ведёт к новым заботам для обеих сторон: для разработчика это дополнительные организационные усилия, для лечебных учреждений – удорожание поддержки. Если справиться с первым – во власти компании, то второе наталкивается на сопротивление со стороны заказчиков.

Иным пользователям вообще кажется странным платить за поддержку: они воспринимают это как оплату за обнаруженные в ходе эксплуатации огрехи программистов. То обстоятельство, что причиной недостатка может быть не разработчик, а несовершенство постановки медицинских задач, заказчик либо не принимает во внимание, либо отказывается понимать, почему он должен платить за кем-то допущенное изначально несовершенство (и здесь есть его правота). Тем более, для ЛПУ будет странным предложение значительно увеличить уже оговоренную плату. В результате, компании сталкиваются со случаями, когда за сопровождение перестают платить.

Выход из такой коллизии надо настойчиво искать. Трудность решений зависит, по крайней мере, от трёх обстоятельств. Первое – это величина платы за сопровождение (о ней – в следующем разделе). Если она небольшая, то её увеличение может быть вполне приемлемым, тогда как попытка нарастить и без того значительную сумму встретит естественный жёсткий отпор. Два других обстоятельства связаны с тем, как понимают цели сотрудничества руководители сторон, точнее, какие цели они ставят перед собою в данный момент. Говоря о стороне пользователя, важно уточнить, кто её представляет. Если это чиновник системы здравоохранения, то высока вероятность, что главным его побуждением будет минимизация затрат. Если в сфере его ответственности МИС уже функционируют, то пусть себе и функционируют, деньги надо беречь. Другое дело, если это главный врач, причём не всякий, а такой, который видит в совершенствовании МИС один из путей заметного улучшения работы своего учреждения. Такой человек способен понять, что незамедлительное удовлетворение запросов и предложений по улучшению МИС имеет прямое отношение к качеству медицинской помощи, а потому стоит денег.

Подчеркнём, что при неблагоприятных обстоятельствах получается следующее. Запросы удовлетворяются с большим опозданием, когда о них уже успевают забыть или

приучаются обходить возникшие препятствия, минуя МИС. Охота о чём-то просить, когда на просьбу так отвечают, пропадает. С недостатками примиряются. Количество запросов перестаёт нарастать, затем снижается. Компания считает, что все довольны. МИС сохраняет в себе те дефекты, которые не являются фатальными для её эксплуатации, к которым приспособились.

Это годится тем, кто стремится не к развитию, не к результату, а к стабильности, к всеобщей успокоенности. Проигрывают здесь врачи, которым надо лечить, проигрывают главные врачи, которые стремятся улучшить медицинскую помощь в рамках своей ответственности, проигрывают компании, которые хотят быть конкурентоспособными на таком рынке, где ценятся прежде всего медицинские функции МИС. Так что тем, кто проигрывает, надо искать выход.

3. ПРАВОВЫЕ, МОРАЛЬНЫЕ И ФИНАНСОВЫЕ АСПЕКТЫ ПОДДЕРЖКИ МИС.

Взаимодействие разработчика с пользователями в процессе производственной эксплуатации МИС – несомненный факт, оно существует, оно необходимо, оно должно существовать. Однако факт и то, что ему уделяется незаслуженно мало внимания. Мы постарались описать содержание этого взаимодействия, показать, каким оно должно быть в правильно построенной работе, что им должно охватываться, каким уровнем сервиса обеспечиваться. Но этого осознания мало для того, что пока складывается самопроизвольно, стихийно. Чтобы сознательно и разумно обеспечивать и развивать взаимодействие, надо определить его природу, основания, из которых оно возникает и на которых строится.

Здесь имеет смысл обратиться к истории медицинских информационных технологий. Первоначально перед ними ставились две простые задачи: взять на себя статистическую обработку данных и помочь врачам в формировании документов и в работе с текстами. Никакого особого взаимодействия ЛПУ с авторами соответствующих технологий не требовалось: получили услугу - заплатите и пользуйтесь.

Но технологии быстро развивались и обнаружили тенденцию проникать в совсем другие сферы. Они стали захватывать всю работу врача с информацией, а это – процесс принятия врачебных решений. Опираясь на работу врача, МИС стали продуцировать информацию не только для медицинских статистиков, но и для руководителей лечебно-диагностического процесса, своевременную, полную, достоверную, то есть пригодную для принятия их управленческих решений. Современные информационные технологии вторглись в лечебно-диагностический процесс, стали его частью, его инфраструктурой, его незаменимым инструментом.

Принципиальное изменение состоит в том, что медицина – не сфера услуг, а непрерывное производство, разработчики ИТ мощно включились в производственный процесс, где уже нельзя обойтись простейшими правилами рынка («заказали – сделали – оплатили», «клиент всегда прав» и т.п.). Заказы обернулись сложными постановками задач, а пользователи – уже не клиенты, а лица, принимающие производственные решения с помощью инструмента, предоставленного разработчиком. Последний теперь волей-неволей воздействует на эти решения, а пользователь, вынужденный сообщать о встретившихся недостатках МИС, тем самым помогает разработчику совершенствовать свой продукт. С

этим взаимопроникновением всё усложнилось, связи утратили временный характер и простоту рыночных отношений,

Никто к такому обороту не готовился, его не предвидели, до сих пор он полностью не осознан, но это ни одну из сторон не освобождает ни от последствий, ни от ответственности за них. Нельзя дальше полагаться на стихийное течение событий. Пора задать ему осознанные и объективно обоснованные ориентиры. Возникла и стала весьма актуальной задача выработать общее представление о взаимных правах и обязанностях, о том, кто, кому, что именно, почему и сколько должен в этих новых обстоятельствах.

3.1. Правовые и моральные аспекты.

Заметим, что взаимозависимость пользователей и разработчиков МИС не симметрична. Разработчик, установив однажды свою систему в ЛПУ и отладив её на этапе ввода, может позволить себе не заботиться о её дальнейшем совершенствовании без видимого ущерба для себя и своей повседневной деятельности. Пользователь попадает в другое положение: если информационная система не следует за его меняющимися потребностями, за развитием медицины, то это поведёт к упущениям в медицинской деятельности, будет вынуждать то тут, то там прибегать к старым, «ручным» способам регистрации и обработки данных (обедняя тем самым электронный архив), станет тормозить работу, снижать качество работы. А эта работа – медицинская помощь людям.

То обстоятельство, что о постоянном взаимодействии в будущем обе стороны не подумали и не договаривались «на берегу», ничего не меняет: радикальные перемены произведены необратимо. Теперь о нормальном течении жизни в новых условиях должны заботиться все, кто их создал. Основанием и стимулом взаимодействия служит, прежде всего, именно моральная ответственность: у разработчика МИС – за условия работы врача с информацией («мы в ответе, за тех, кого приручили»), у медиков – за то, чтобы новую технологию наилучшим образом использовать в работе с пациентами. На возможные замечания, что здесь «бизнес и ничего личного», можно уверенно возразить, что в данном случае мы имеем дело с социально ориентированным бизнесом, а это означает более высокую ответственность за последствия.

Следует заметить, что вопрос о том, кто, кому и почему должен, не так прост, как кажется на первый взгляд. Своими запросами пользователи, по сути, осуществляют до-тошную экспертизу МИС и способствуют повышению качества программного продукта, а значит, улучшают его репутацию, что для компании не безразлично. Заслуживает внимание и такое соображение: почему, собственно, пользователю приходится обращаться со своими просьбами? Не потому ли, что разработчик чего-то недоделал? На это можно по-

лучить разные ответы: кто-то скажет, что раньше надо было обо всём заботиться и хорошо ставить задачи. Где-то будет уместным возражение, что задачи ставили вовсе не те, кто сейчас вынужден использовать МИС, те же, кому в своё время пришлось их формулировать, очень смутно представляли себе полный перечень требований и возможности автоматизации для ЛПУ, то есть руководствовались поверхностным пониманием лечебно-диагностического процесса как объекта автоматизации. Всё справедливо, все без вины виноваты. Но в итоге крайними оказываются те, кому приходится повседневно действовать в уже созданной информационной среде, современной, но далеко не совершенной: разработчик со своим коллективом, которым нужно платить зарплату и налоги, лечащие врачи, которым надо вводить, хранить, получать и передавать информацию, и, в конечном счёте, пациенты. При решении всех проблем взаимодействия ЛПУ и разработчика это надо держать в уме.

3.2. Затраты разработчика на поддержку МИС.

При всей несомненной моральной ответственности перед пользователями, у разработчика есть совершенно строгая юридическая ответственность в своей собственной сфере: надо своевременно и с учётом квалификации оплачивать труд сотрудников, платить налоги, оплачивать аренду помещений, обеспечивать экономическую эффективность производства. Заботу о взаимодействии с пользователями МИС разработчик может проявлять в той мере, в какой её удаётся сочетать со всей остальной деятельностью компании. Нельзя сбрасывать со счетов, что разработчик МИС – это коммерческая компания, работающая в юрисдикции российского права.

Мы описали тот уровень ожиданий и требований к разработчику, который обусловлен современным пониманием автоматизации лечебно-диагностического процесса, изложили, каким взаимодействие разработчика и пользователей должно быть, каким оно может быть в принципе. Но сегодняшняя картина в этой сфере деятельности складывалась не на основе нынешних представлений, не по плану, а стихийно, как и вся информатизация здравоохранения. Поэтому фактическое положение не соответствует желаемому. Это означает, что затраты компании на поддержку МИС нельзя оценивать на основе описанного идеала, они другие, как правило, они выше чем могли бы быть, если бы всё начать сначала при нынешнем понимании проблем информатизации.

В такой ситуации остаётся одно: принять существующее положение как данность, старательно его описать, понять, с чем можно мириться, а что требует перемен, какими бы

трудными они ни были, а затем упорно и долго, шаг за шагом двигаться от действительно-го к желаемому.

Чтобы полноценно раскрыть все сложности организации сопровождения МИС, опишем ту ситуацию с затратами, которая объективно складывается у любого разработчика в настоящее время. В основе производственной деятельности компании лежит работа двух связанных между собою служб. Одна – это служба разработки МИС, включая аналитиков, специалистов в предметной области (консультантов), разработчиков программ, тестировщиков, специалистов по контролю качества и документированию, Другая - служба технической поддержки, включая, как правило, выделенного руководителя и специалистов.

Затраты компании разработчика имеют вполне определенную структуру:

- заработная плата сотрудников – та сумма, которую каждый сотрудник получает на руки (основная статья затрат любой ИТ-компании);
- налог на заработную плату (НДФЛ) и страховые взносы в бюджеты Пенсионного фонда Российской Федерации и Фонда социального страхования Российской Федерации, составляющие вместе в общем случае 50% от заработной платы;
- аренда помещений;
- инфраструктурные затраты (телефония, интернет, работа сайтов и т.д.);
- командировочные расходы;
- расходы на обучение сотрудников;
- налоги на деятельность компании разработчика.

Практически все эти затраты являются постоянными. Поэтому организация работы компании по сопровождению МИС должна выполняться с учетом того, что все возникающие затраты надо компенсировать. Кроме того, должна быть обеспечена заданная норма прибыли (маржинальность).

Чем больше различных функций и возможностей ожидается от МИС, чем выше требования к качеству предоставляемых услуг по сопровождению, тем выше требования к профессионализму, знаниям и количеству персонала. Тем, соответственно, выше затраты компании, в том числе – на выполнение обязательств по сопровождению внедрённых МИС.

На сегодня рынок труда для ИТ-специалистов в различных регионах разный. В крупных городах, таких как Москва и Санкт-Петербург, средние зарплаты ИТ-специалистов максимальны. В регионах они меньше, но не намного. В общем, ориентировочно можно принять средний уровень заработной платы ИТ-специалиста компаний - раз-

работчиков МИС за 50 тыс. руб./месяц или 312 руб. / человеко-час на руки или с учетом налога НДФЛ и страховых взносов – 75 тыс. руб./месяц или 469 руб. / человеко-час .

Теперь рассмотрим формирование стоимости процесса сопровождения МИС с точки зрения затрат разработчика. Например, разработчик получает от администратора МИС лечебно-профилактического учреждения запрос на изменение системы. Его обработка означает следующие затраты.

- Регистрация запроса, его рассмотрение, попытка повторить все это в системе, уточнение задачи у администратора МИС ЛПУ и итоговое оформление задачи для разработчика – 30 минут чистого времени в среднем на один запрос (хотя вся работа могла продолжаться в течении дня, мы учитываем, что в течение рабочего дня специалист решает не одну задачу, а много). Стоимость регистрации запроса в этом случае составит 234,5 руб.
- Время, затрачиваемое программистом на выполнение изменений по зарегистрированному запросу, колеблется очень широко, от нескольких минут до нескольких дней. В среднем примем его за 4 человеко-часа, что составит 1876 руб. Затраты на тестирование составляют, как правило меньше, чем разработка, поэтому оставим на эту работу 1 человеко-час, что составит 469 руб.
- Затраты на документирование: описание выполненного изменения, внесение доработок в инструкцию пользователя, а нередко и администратора – примем за 20 минут, что составит 156 руб.

Таким образом, только прямые затраты на заработную плату сотрудников с учетом налога НДФЛ и страховых взносов уже составили в этом примере 2735,5 руб. за удовлетворение одного запроса. К этой сумме нужно прибавить накладные расходы, связанные с работой управленцев (примерно 15%), а также затраты на аренду помещений, электро-снабжение, амортизацию и другие сопроводительные расходы (их тоже можно принять за 15%). Таким образом, сумма выросла до 3556,2 руб. Оказание услуг коммерческой компании не может выполняться без прибыли, поэтому учтем норму прибыли, которая может быть очень разной, но для определенности примем ее за 20%. Итак, стоимость обработки запроса составила 4267,4 руб. Сверх этой суммы нужно ещё учесть налоги, которые платит компания со своей деятельности.

Компания-разработчик, в которой работает один из авторов, имеет такую статистику: при поддержке МИС в определенном лечебном учреждении, в среднем, требуется обслуживание 10-20 запросов в месяц, из них 5-15 требуют внесения изменений в МИС. При поддержке регионального проекта количество запросов составляет от 40 до 80, а количество изменений 40-50. Однако весь этот объем приходится не на одно лечебное учрежде-

ние, а сразу на все региональные ЛПУ. В результате, средние затраты разработчика на поддержку одного лечебного учреждения по описанной выше методике расчета составляют порядка 40 тыс. руб. в месяц.

В реальности стоимость услуг по поддержке не рассчитывается для каждого отдельного случая. Все риски экономической эффективности, включая получение прибыли или убытков от поддержки МИС, лежат целиком на компании-разработчике. Поэтому компании вынуждены на свой страх и риск определять стоимость этой услуги для заказчиков, исходя не из точного расчета затрат, а из собственного опыта: сколько примерно запросов приходит от одного заказчика в течение месяца, сколько при этом из них разбираются на уровне отдела сопровождения и не требуют доработок, а сколько поступают программистам и вынуждают вносить изменения в МИС.

Чем больше проект, особенно если речь идет о региональных проектах, тем выше затраты, т.к. они, в основном, зависят от числа запросов в месяц и средней стоимости удовлетворения одного запроса. Можно допустить, что средняя стоимость – величина постоянная на протяжении длительного времени.

Иначе обстоит дело с числом запросов. Сразу после ввода МИС в эксплуатацию оно обычно невелико, спустя несколько недель начинает нарастать, особенно тогда, когда врачи понимают, что на их запросы реагируют благожелательно, конструктивно и оперативно, а затем, после того, как разработчик справился с этой волной, количество запросов снижается: в основном, запросы касаются только сравнительно редко используемых функций. Иное дело, если вмешиваются внешние факторы: выходят новые приказы, меняются правила сдачи реестров или требуются интеграции с другими системами, например, с новыми версиями федеральных сервисов ЕГИСЗ. Внешняя среда вызывает новый поток запросов на изменение и, таким образом, потребность в сопровождении МИС не прекращается до тех пор, пока система эксплуатируется в ЛПУ.

Из сказанного можно сделать 3 любопытных вывода.

1) Ясно, что количество запросов зависит от состояния МИС к началу внедрения. При этом, поскольку поправки, внесенные по запросам одного ЛПУ, распространяются и на другие ЛПУ, то при вводе МИС в другом учреждении запросов станет меньше. Значит ли это, что учреждения, внедрившие у себя МИС, уже эксплуатировавшуюся в других ЛПУ, должно платить за сопровождение меньше? Или иначе: если да, меньше, то почему первое учреждение платило больше? За что оно платило?

Расширим проблему. Если одна МИС требует на первом году эксплуатации много поправок, а другая мало, разумно ли, что пользователи должны платить за поддержку первой МИС больше на том основании, что больше затраты ее разработчика? Не логичнее ли

требовать совершенно обратную ситуацию: чем больше система требует доработок по сравнению с другими решениями, тем хуже ее изначальное качество и тем меньше должна быть ее стоимость?

Обратим еще раз внимание, что требования пользователей имеют первоосновой не их личные желания, а стремление сделать удобной свою работу, работа же у всех одна и та же, медицина с её правилами. Таким образом, получается, что пользователи оплачивают доведение МИС до необходимых всем кондиций тем больше, чем меньше это сумел сделать разработчик. Менее совершенна МИС – больше запросов – больше плата за сопровождение. Противоречие налицо. Его можно ослабить, требуя предоставления гарантийного срока, когда недоделки устраняются за счёт компании. Такая практика выглядит вполне оправданной. Она заставляет рынок нести ответственность за качество своих систем. Но устранить совсем описанное нами противоречие вряд ли возможно.

2) Поправка в МИС, сделанная с подачи одного учреждения, передаётся всем. Затраты на неё такие же, как если бы этих всех не было вообще. Но за поддержку платят все. Значит, стоимость поддержки МИС не должна рассчитываться таким образом, чтобы одно лечебное учреждение полностью оплачивало затраты на доработку программного обеспечения. Решение сводится к тому, что стоимость сопровождения МИС для отдельного ЛПУ должна быть меньше, чем стоимость всех затрат компании-разработчика МИС на работу службы сопровождения, так как каждая поправка делается сразу для всех клиентов компании, а не для одного.

3) Число запросов может увеличиваться при их неполноценном удовлетворении. А при несвоевременном удовлетворении могут поступать из разных учреждений одни и те же запросы, так что возникают ещё и проблемы с дублями. Подобные примеры показывают, что количество запросов определяется ещё и тем, как организовано их удовлетворение (квалификация программистов, специальные программы, конструктор, средства тестирования). Лучше организация – меньше повторных запросов – меньше затрат. Это означает, что компания-разработчик может и должна обеспечивать свою экономическую эффективность не только за счет высокой цены на сопровождение, но и собственными усилиями: совершенствованием процесса сопровождения и разработки МИС, лучшим качеством работы.

Итак, затраты компании-разработчика на поддержку системы определяются целым комплексом факторов: количеством запросов от пользователей, количеством ЛПУ и пользователей, находящихся на обслуживании, внешними причинами, такими, как изменения законодательства и порядков информационного обмена с территориальными фондами

ОМС, страховыми компаниями или федеральными сервисами ЕГИСЗ, а также собственной зрелостью и ответственным отношением к делу.

3.3. Сколько стоит поддержка МИС для ЛПУ.

Вопрос конечной стоимости сопровождения МИС для заказчика является крайне сложным. На сегодняшний день все разработчики присутствуют на открытом рынке, поэтому цена услуг и программных продуктов в существенной степени регулируется законами рынка. Невозможно просто подсчитать собственные затраты разработчика, учесть норму прибыли и выставить полученную цену. Необходимо учитывать различные риски ведения бизнеса, особенно если это касается государственного сектора и государственных закупок. всё это каждый разработчик вырабатывает для себя индивидуально, по собственным правилам. Поэтому в данной работе мы не можем предложить некую рекомендованную цену, которая бы являлась оптимальной и для разработчика и для заказчика,

Тем не менее, можно привести среднерыночные данные по результатам мониторинга рынка за 2013 г. (А.В.Гусев, "Обзор государственных закупок на программное обеспечение информатизации здравоохранения в 2013 г.", Выставка и конференция "Medsoft 2014", 25— 27 марта 2014 г., Москва.: http://www.kmis.ru/site.nsf/pages/medsoft2014_2.htm).

Согласно этим данным, мы имеем следующее представление о среднерыночной стоимости сопровождения МИС:

- Стоимость сопровождения одной МИС в год на регион составила 19,4 млн. руб.
- Средняя стоимость этой же услуги за весь региональный фрагмент ЕГИСЗ в 2013 г. составила 26,9 млн. руб.
- Средняя стоимость сопровождения МИС для одной медицинской организации колеблется очень широко: от нескольких тысяч до 200-350 тыс. руб.

Стоимость эта существенно зависит как от количества медицинских организаций в регионе и количества пользователей в них, так и от конкретного бренда той МИС, на которую поддержка закупается. Чем больше функциональных возможностей в системе, тем, как правило, стоимость сопровождения выше. Обычно закупка сопровождения осуществляется госзаказчиками на год вперед.

Безусловно, приведенные средние цифры не являются доступными всем ЛПУ. В этой связи компании-разработчики, как правило, стремятся в ходе переговорного процесса подобрать такой уровень услуг, который был бы приемлем ЛПУ по стоимости.

*

Заканчивая обсуждение сегодняшнего положения с затратами компании и лечебных учреждений на поддержку эксплуатируемых МИС, подчеркнём ещё раз, что оно сложилось стихийно и в ряде отношений должно меняться, как и вообще организация взаимодействия разработчиков и пользователей. Сам по себе рынок вряд ли способен вызвать необходимые перемены. Предстоит найти способы, которыми этот рынок можно было бы рационально регулировать.

ЗАКЛЮЧЕНИЕ.

ВЗАИМОДЕЙСТВИЕ РАЗРАБОТЧИКА И ПОЛЬЗОВАТЕЛЕЙ МИС – АНТИЭНТРОПИЙНАЯ СИСТЕМА.

По старому правилу программистов «Всё, что портится, портится. То, что не портится, портится тоже». Всё устаревает и разрушается, таков закон энтропии. Взаимодействие разработчика и пользователей МИС препятствует старению последней – это антиэнтропийная система управления, целеустремлённая, открытая через МИС во внешний мир, в живой мир медицины.

Цель этой системы – поддерживать соответствие МИС развивающимся информационным потребностям ЛПУ. Коллективный субъект управления – лечащие врачи и медицинские статистики всех ЛПУ, где используется МИС разработчика. Объекты управления – программисты, вносящие изменения в МИС по запросам пользователей. Аналитический блок представлен теми сотрудниками ЛПУ и компании-разработчика, которые проверяют правильность запроса, полноту и сроки его удовлетворения. Налицо совокупность прямых и обратных связей. Чётко выделяются подсистемы для основных функций – формирования запроса, внесения изменений в программу, обеспечения двусторонней связи между пользователем и разработчиком. На обеих сторонах есть специальные инструменты, предназначенные именно для взаимодействия. В результате деятельности системы создаётся электронная база данных о запросах, удовлетворённых, отвергнутых и находящихся в работе.

Важно отметить своеобразие этой системы. В ней функционально объединены структуры и сотрудники учреждений, административно друг другу не подчинённых. Это вынуждает к раздвоенности аналитического блока и базы данных о запросах, а также создаёт неопределённость в вопросах права и финансовых отношений. Вместе с тем обстоятельством, что вся система включает в себя на разных уровнях людей, принимающих решения, это делает чрезвычайно важными все вопросы организации: установление правил, контроль за их соблюдением, выбор критериев, позволяющих оценивать эффективность взаимодействия. От того, как эти вопросы будут решаться, зависит степень достижения поставленной цели и способность самого взаимодействия к совершенствованию.

С выработкой чётких и исчерпывающих представлений о взаимодействии разработчика и пользователей МИС мы опаздываем. Информатизация здравоохранения наращается в масштабах страны, при любых вариантах в её основе всегда будет функциони-

рование информационных систем в лечебных учреждениях, работа лечащих врачей в новых условиях. По мере того, как врачи всё больше будут вникать в эти условия, всё больше понимать, что вообще могут дать им информационные технологии, их требовательность к МИС будет возрастать. Нельзя сказать, чтобы разработчики были достаточно готовы к такому развитию событий.

Надо наверстывать упущенное. Для этого мало обсуждать проблему взаимодействия вообще. Необходимо показывать и учитывать, как она решена в конкретных системах, которые предлагаются на рынке, на выставках, конкурсах МИС и т.д. Надо взять за правило при сравнительных оценках МИС, помимо их технических характеристик и медицинских функций, использовать в качестве критериев и те признаки, которые позволяют понять, как предусмотрено взаимодействие пользователей с разработчиком МИС: использует ли разработчик автоматизированную систему регистрации запросов пользователей и их удовлетворения, имеется ли полноценное «Соглашение об уровне предоставления услуги» (SLA), каковы ориентировочные сроки удовлетворения запросов. Тогда вся проблема будет постоянно находиться в поле зрения всех, кто имеет отношение к информационным технологиям для лечебных учреждений.