

ОБРАБОТКА ИНФОРМАЦИИ О ПОЛЬЗОВАТЕЛЯХ В ПРОГРАММНОМ КОМПЛЕКСЕ УПРАВЛЕНИЯ ПЕРЕВОДОМ

Кириянов А. А., Сироткин В. Ю., Анашкин Р. В.

ООО «Связь-Строй» (121614, г. Москва, Осенний бульвар, д. 18, корп. 2, офис 428, e-mail: 9888160@mail.ru)

В статье рассмотрены вопросы, связанные с разработкой автоматизированной системы управления переводом, предназначенной для пользователей сети интернет и предоставляющей им возможность получить языковой перевод контента сайтов, выполненный переводчиками. Указанная система обеспечивает предоставление он-лайн интерфейса переводчика с поддержкой памяти переводов и обсуждение перевода по сегментам, а также бесплатно предоставляет интерфейс переводчика с глоссарием. Система реализована в виде программного комплекса (ПК), функционально-алгоритмическая структура которого основана на взаимодействии входящих в него модулей. В частности, предметом рассмотрения настоящей статьи является функционал модуля «Клиент», реализующий схему авторизации пользователей в ПК, алгоритмические и программные решения, позволяющие организовать ввод, обработку, хранение и вывод информации о пользователях ПК.

Ключевые слова: программное обеспечение, память переводов, сервис управления контентом, локализация сайтов.

PROCESSING INFORMATION ABOUT USERS IN THE TRANSLATION MANAGEMENT SOFTWARE PACKAGE

Kiryaynov A. A., Siroykin Y. V., Anashkin R. V.

ООО "Svyaz-Stroy" (Office 428, Bidg. 2, 18, OsenniyBulvar, 121614, Moscow, Russia 9888160@mail.ru)

The questions related to the development of automated translation management system designed for Internet users and provides them with the opportunity to obtain language translation of content sites, made by translators. Said system ensures the provision of on-line interpreter interface with support for translation memory and translation discussion by segment, as well as free interpreter provides an interface to the glossary. The system is implemented as a software package (PC) functional algorithmic structure of which is based on the interaction of its constituent modules. In particular, the subject of this paper is a functional module "Client", which implements user authentication scheme in the PC, algorithmic and software solutions allowing to organize the input, processing, storage and display information about users of the PC.

Key words: software, translation memory, content management services, website localization.

Введение

В настоящее время востребованной является задача перевода контента сайтов на различные иностранные языки. Как ответ на эту потребность появились сайты, организующие работу по таким переводам [1, 2]. Но как выяснилось, подобные сайты имеют схожие недостатки, такие как невозможность предоставить онлайн интерфейс переводчика и обеспечить работу с памятью переводов, а также организовать обсуждение перевода по сегментам.

Целью исследования стала необходимость разработать программный комплекс (ПК) управления переводом, освобожденный от этих недостатков и обеспечивающий ввод, обработку, хранение и вывод информации о пользователях.

Материалы и методы исследования

В связи с этим разработан ПК [3], предназначенный для управления процессами перевода контента и ввода, обработки, хранения и вывода информации о пользователях интернет, которые регистрируются в ПК.

Непосредственно ввод, обработку, хранение и вывод осуществляет модуль «Клиент», выполняющий такие функции, как: добавление карточки клиента, в том числе:

- 1) ввод названия клиента;
- 2) ввод Ф.И.О., должности, прав доступа для каждого из контактных лиц клиента;
- б) ввод контактной информации о каждом из контактных лиц клиента;
- в) указание предпочтительного способа оплаты: Moneybookers; Qiwi; Webmoney; Ассист; Банковские квитанции; Безналичный расчет; Моби; Яндекс.деньги;
- г) ввод адреса клиента, юридических лиц контрагентов со стороны клиента, их реквизитов и валюты расчета;
- д) указание тематики, отражающей предметную область (выбор из перечня);
- е) указание индивидуальной настройки (если необходимо) хранения материалов на файл-сервере или FTP-сервере;
- ж) набор телефона контактного лица клиента в случае использования VoIP-телефонии;
- и) выбор тарифного плана (из перечня);
- к) выбор схемы тарификации с учетом повторов:
 - 1) тарификация по словам с учетом скидок за повторы;
 - 2) выбор шаблона скидок за повторы из перечня шаблонов.

При этом модуль «Клиент» обеспечивает:

- а) обращение к БД при формировании вывода данных для загрузки ранее введенной в БД информации о клиенте;
- б) формирование содержимого HTML сообщения для вывода;
- в) получение данных от браузера клиента по протоколу HTTP;
- г) обращение к БД при получении данных, введенных клиентом с целью сохранения или изменения ранее сохраненных данных;
- д) проверка на соответствие формату введенных клиентом данных с помощью алгоритмов верификации и вывод информации об ошибке в случае определения таковой путем вывода текста ошибки в браузер клиента в формате HTML.

Функционал модуля «Клиент» создан на основе плагина для фреймворка Django [4,5] «Userena», для чего этот модуль интегрирован в ПК и из него заимствованы такие функции, как регистрация, вход в аккаунт, выход из аккаунта, а отсутствующие в нём функции, специфичные для ПК (такие как редактирование услуг переводчика или указания

предпочитаемого пользователем способа оплаты), созданы дополнительно. На рис. 1 изображена блок-схема системы авторизации модуля «Клиент».

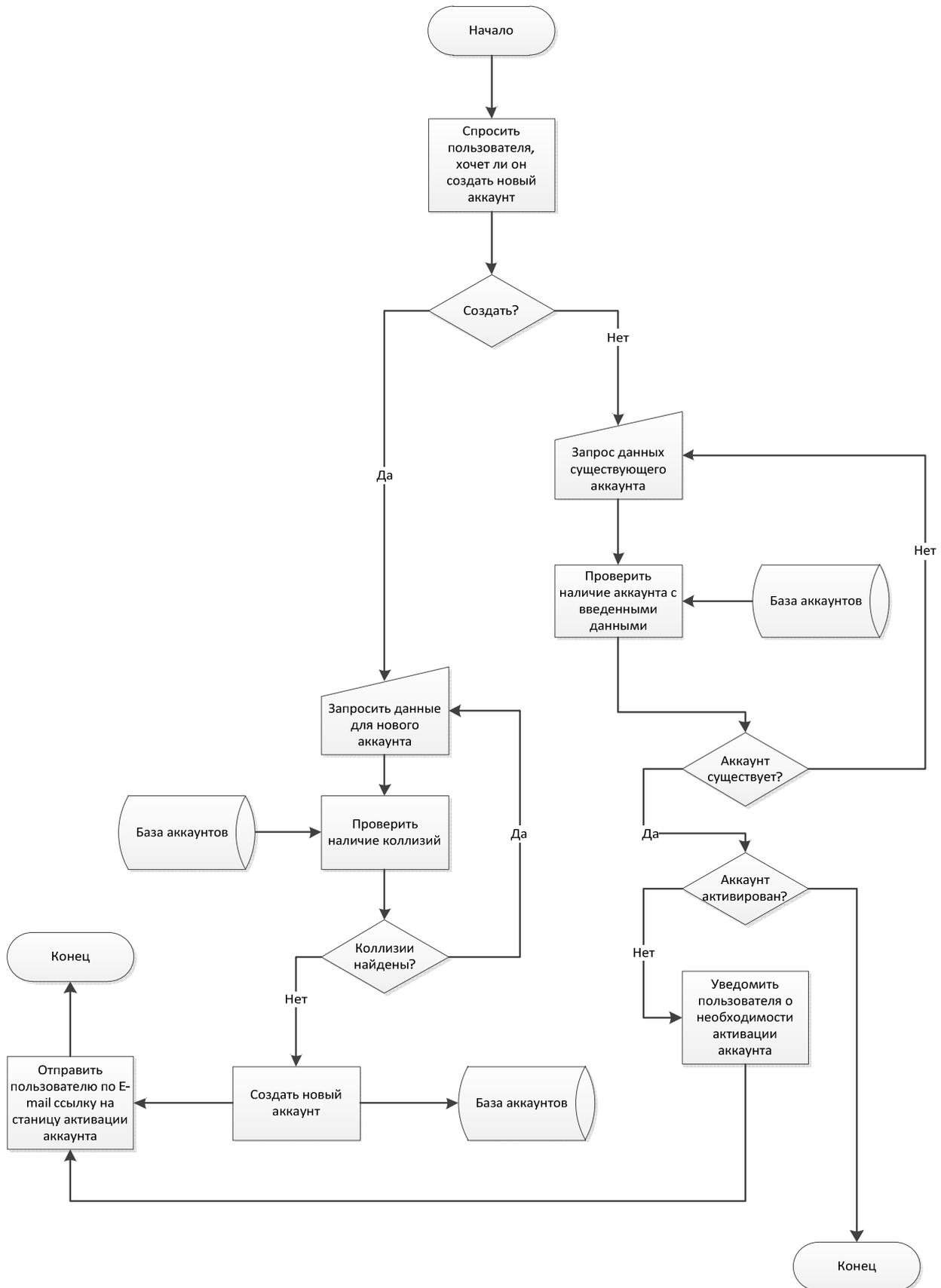


Рис. 1. Блок-схема системы авторизации модуля «Клиент»

Данная схема авторизации работает следующим образом: при входе пользователя на сайт он может или войти в аккаунт, или зарегистрировать новый. Если аккаунт существует, но не активирован, пользователь получает инструкцию по активации. При неудачной попытке входа пользователь уведомляется о некорректности данных. В этом случае пользователь может либо попытаться войти снова, либо зарегистрировать новый аккаунт. Регистрация представляет собой html-форму ввода со стандартными полями ввода. В ней реализована проверка на наличие коллизий введенных данных с уже существующими (это необходимо для обеспечения уникальности имени пользователя и т.п.).

Безопасность пользовательских данных и системы в целом обеспечивается такими методами, как: отказ от хранения непосредственно пользовательского пароля (хранения только его хеш-кода), необходимость активации аккаунтов через электронную почту (что резко снижает возможности по автоматической регистрации вредоносных программ-ботов), разграничение пользователей по группам с различными правами доступа (позволяет уменьшить риск умышленной порчи данных) и т. д.

Модуль «Клиент» взаимодействует с рядом подсистем ПК. На блок-схеме (рис. 2) показано, как в зависимости от роли пользователя в системе, графический пользовательский интерфейс может иметь дополнительные пункты в основном навигационном меню (например, если ролью пользователя является «ИСПОЛНИТЕЛЬ-ПЕРЕВОДЧИК», то в меню появится дополнительная вкладка «Переводчик» и т.п.).

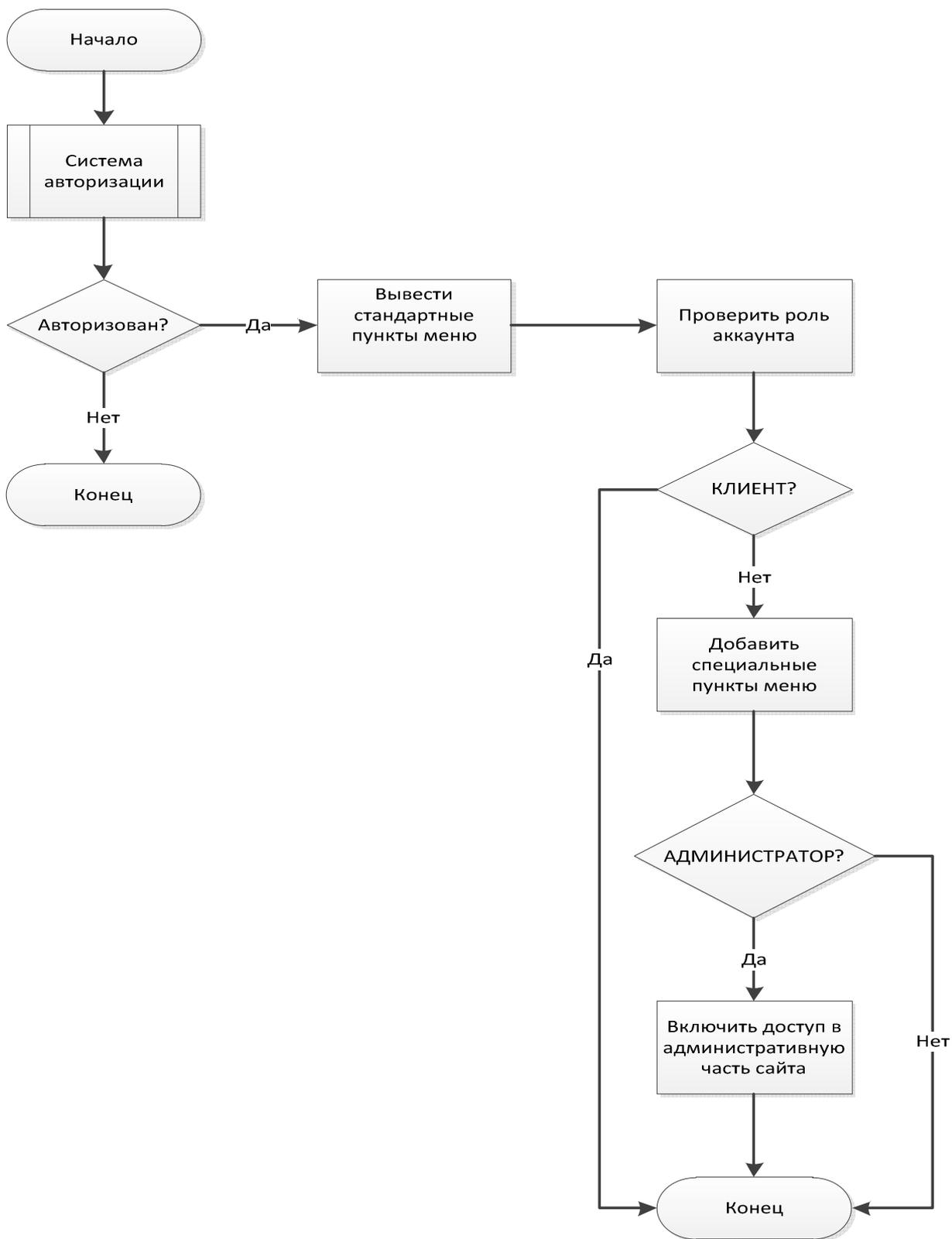


Рис. 2. Зависимость графического пользовательского интерфейса от роли пользователя в системе

Пользователь использует модуль «Клиент» при регистрации в ПК. Это происходит следующим образом: при нажатии на кнопку «регистрация» сервер генерирует форму ввода

со стандартными полями, основанную на классе ORM-модели. Т. к. создаётся новый аккаунт, то обращения к базе данных на этом этапе не происходит. После заполнения всех полей и успешной их валидации пользователю отправляется сообщение по e-mail со ссылкой на страницу активации аккаунта.

После успешной активации пользователь получает возможность ознакомиться с остальным функционалом модуля «Клиент». При нажатии на кнопку «профиль» аккаунта, пользователь попадает на страницу, где в зависимости от установленной роли может выбрать несколько различных действий, связанных с обновлением и/или изменением текущего аккаунта. Все web-страницы с этими действиями имеют формат URL вида «/accounts/[username]/[action]/», где [username] – имя активного аккаунта, а [action] – идентификатор выбранного действия над аккаунтом. Такая гибкость формата URL обеспечивается за счёт мощного инструмента фреймворка «Django» – роутинга на основе регулярных выражений (regex) [4].

Вне зависимости от установленной роли пользователю обеспечен доступ к 4 базовым действиям над аккаунтом: просмотр, изменить, сменить пароль, сменить email.

В ответ на запрос каждого из этих действий сервер формирует web-страницу с html-формой со стандартными полями ввода на основании соответствующих действию атрибутов класса ORM-модели User (пользователь). Валидация данных, как правило, осуществляется при помощи регулярных выражений (regex), а также проверки типа введенных данных, которые автоматически выводятся из типа данных соответствующей ORM-модели. Валидация, в целях повышения надежности системы, проводится как на стороне клиента, так и на стороне сервера.

Действие «Просмотр» позволяет пользователю проверить данные текущего активного аккаунта. Web-страница, соответствующая этому действию, формируется посредством выведения загруженных из базы аккаунтов данных. При этом также происходит обращение к сервису gravatar.com, откуда, при наличии у текущего активного аккаунта учётной записи в этом сервисе (что определяется по полю «e-mail»), модуль получает изображение-аватар, связанный с этим электронным адресом.

Действие «Изменить» позволяет редактировать и/или дополнять данные текущего активного аккаунта (за исключением служебной информации). Полями для изменения в этом действии являются: Ф.И.О., уровень приватности аккаунта, пол, телефон, факс, адрес и дополнительные данные. Также есть возможность выбрать файл изображения с пользовательского компьютера, который будет использоваться в html-форме действия «Изменить».

Результаты исследования и их обсуждение

На рис. 3 представлена блок-схема обобщенного алгоритма генерации и обработки html-форм для различных действий (алгоритм валидации данных на стороне клиента опущен, т.к. он актуален при условии, что в браузере пользователя отключен javascript).

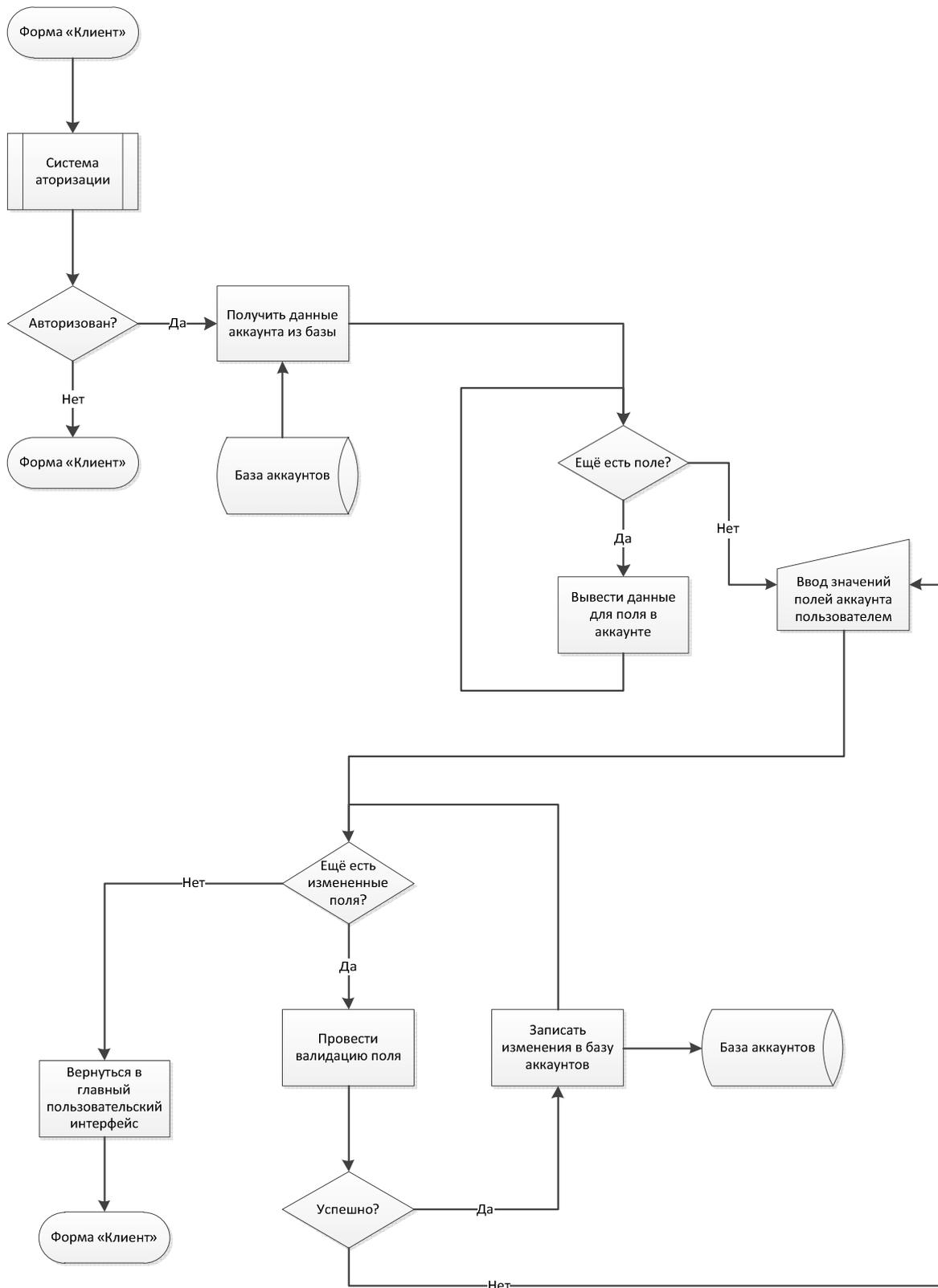


Рис. 3. Обобщенный алгоритм генерации и обработки html-форм

Различные модификаторы аккаунтов могут давать доступ к дополнительным возможностям редактирования. Например, роль «ИСПОЛНИТЕЛЬ-ПЕРЕВОДЧИК» даёт возможность редактировать профиль переводчика, а флаг «corporate» в классе ORM-модели User (пользователь) позволяет выбрать, кем является клиент – физическим или юридическим лицом, и даёт возможность указывать ряд дополнительных уникальных параметров.

Программный модуль «Клиент» реализует функционал по добавлению, хранению, систематизации и редактированию информации об аккаунтах пользователей ПК, осуществляет менеджмент ролей аккаунтов, реализует функции по аутентификации и авторизации пользователей в ПК, а также обеспечивает целостность и персистентность хранимых данных, используя для этого ORM-модели фреймворка Django. Также информация из этого модуля используется для параметризации поведения многих подсистем ПК.

Ввод данных происходит следующим образом: от ПК в браузер клиента принимаются поля для ввода, передача которых осуществляется путём HTTP-запросов формата GET и POST. После того как с помощью запросов поля отображаются у клиента в браузере, поля для ввода данных заполняются пользователем.

Входными данными модуля «Клиент» являются данные регистрации личного кабинета в ПК, переданные клиентом. Состав входных данных зависит от того, является ли клиент юридическим или физическим лицом.

Ввод данных представляет собой вывод формы для заполнения, имеющей стандартные поля для ввода пользовательских данных и разных выборов, где клиент может отобрать нужные ему данные, переданные в браузер пользователя в формате стандартного языка разметки HTML (HyperTextMarkupLanguage). Входные и выходные данные представляют собой текстовые строки, хранящиеся в кодировке UTF-8, которые содержат команды и результаты выполнения.

Данные структурированы в таблицах БД:

- 1) Permission – Права доступа;
- 2) Group – Группы;
- 3) User – Пользователь;
- 4) UserObjectPermission – Права доступа к объекту;
- 5) GroupObjectPermission – Права группы для доступа к объекту;
- 6) UserProfile – Профиль пользователя;
- 7) Passport – Паспорт пользователя [опционально, по желанию пользователя];
- 8) Company – Карточка компании.

Заключение

Таким образом, обеспечивается ввод, обработка, хранение и вывод информации о пользователях в программном комплексе управления переводом.

Работа выполняется при финансовой поддержке Министерства образования и науки РФ (государственный контракт № 07.524.11.4020)

Список литературы

1. Сайт Transifex.com [Электронный ресурс]. – Режим доступа: <https://www.transifex.com/about/> (дата обращения 27.05.2013 г.).
2. One Hour Translation. Ведущая служба профессионального перевода [Электронный ресурс]. – Режим доступа: <http://www.onehourtranslation.com/translation/about-us> (дата обращения 27.05.2013 г.).
3. Сайт Translateok.com [Электронный ресурс]. – Режим доступа: <http://translateok.com/about/> (дата обращения 27.05.2013 г.).
4. Djangodocumentation. EverythingyouneedtoknowaboutDjango [Электронный ресурс]. – Режимдоступа:<https://docs.djangoproject.com/en/1.5/> (дата обращения 28.05.2013 г.).
5. Userena Introduction. – Режим доступа: <http://docs.django-userena.org/en/latest/settings.html> (дата обращения 28.05.2013 г.).

Рецензенты:

Марсов Вадим Израилевич, д-р техн. наук, профессор кафедры «Автоматизация производственных процессов» Московского автомобильно-дорожного государственного технического университета, г. Москва.

Суминов Игорь Вячеславович, д-р техн. наук, профессор, проректор по научной работе, ФГБОУ ВПО «МАТИ-РГТУ» им. К. Э. Циолковского, г. Москва.