

способы решения необходимой пользователю задачи при выполнении последовательности действий, заложенной в бизнес-процесс ПС;

7) принцип мостовых перил либо «защита от дурака» – данные необходимо защищать от случайных действий пользователя и на уровне прикладных ПС;

8) принцип единства – настройки и элементы управления необходимо группировать, а не скрывать в отдельных разделах.

UI (User Interface, пользовательский интерфейс) – все компоненты системы, которые предоставляют пользователю информацию и являются инструментами управления для выполнения определенных задач [3]. UI-дизайн – это этап детального проектирования внешнего вида ПС и всех его элементов. По завершению этапа UI-дизайн формируется готовый дизайн ПС с доскональными инструкциями для реализации интерфейса разработчиками.

На этапе UI-дизайна необходимо придерживаться следующих принципов [2]:

1) элементы управления должны быть привычны – использование универсальных элементов и визуальных образов при разработке ПС позволит лучше адаптироваться пользователю и перенести приобретенный опыт;

2) убираем очевидное – не стоит выделять очевидные элементы интерфейса, нужно сосредоточиться только на важных для функциональности элементов интерфейса;

3) однородность – поддержка однородности элементов интерфейса, стиливого оформления, синтаксической и семантической согласованности;

4) принцип группировки – информацию желательно разбивать на логические блоки (группы), так пользователю легче ориентироваться;

5) все полезное на виду – важные элементы интерфейса необходимо размещать «на виду» и выделять соответствующим образом;

6) люди не читают, а просматривают – при проектировании интерфейсов необходимо использовать короткие словосочетания, т.к. пользователи не любят читать большие массивы текстов.

Вышеописанные принципы UX- и UI-дизайна не всегда возможно соблюдать, поэтому приходится возвращаться на предыдущие этапы разработки, и переделывать или оптимизировать неудачные решения. Огромную роль играет тестирование удобства интерфейса на фокус-группах, и сбор статистики с альфа- и бета-тестирования. Такие исследования помогают выявить основные проблемы, с которыми сталкиваются пользователи, и, по возможности, их устранить.

Список использованных источников:

1. Копов С.А., Шибанов С.В., Макарычев П.П. User Experience как новый уровень качества программного обеспечения // Труды Международного симпозиума «Надежность и качество»: докл. Междунар. науч. конф., Пенза, 19 марта 2009. – Пенза : Пензенский государственный университет, 2009. – 19 с.

2. Памятка UX / UI дизайнеру. Принципы построения интерфейсов. [Электронный ресурс].– Режим доступа: https://habrahabr.ru/company/SECL_GROUP/blog/182208. Дата доступа 06.04.2017.

3. ГОСТ Р ИСО 9241-210 – 2012. Эргономика взаимодействия человек-система.

МОБИЛЬНОЕ ПРИЛОЖЕНИЕ «МЕНЕДЖЕР ПЕРИОДИЧЕСКИХ ПЛАТЕЖЕЙ»

Институт информационных технологий БГУИР, г.Минск, Республика Беларусь

Бердник А.В.

Матвеев А.В. – ассистент

В докладе представлены результаты разработки мобильного приложения, позволяющего производить оплаты за товары и услуги через мобильные устройства. В мобильном приложении обеспечена защита данных, организован удобный интерфейс.

В настоящее время можно произвести оплату за покупки товаров, не выходя из дома, через Интернет. Не стали исключением и необходимые многим периодические платежи, такие как оплата телефона, учебы или услуг ЖКХ. Некоторые платежные системы в нашей стране позволяют настроить автоматические платежи на некоторые услуги, однако функционал регулирования частоты оплаты, как правило, отсутствует, и в целом данный способ не всегда уместен либо пугает некоторых пользователей из-за отсутствия личного контроля за платежами. Поэтому необходимо приложение, которое будет в себе совмещать и удобства автоматической оплаты услуг, и контроль пользователем периодичности, сроков оплаты, необходимых услуг и уровня автоматизации приложения.

В спроектированном интерфейсе пользователь сможет создавать несколько видов событий и настраивать как ему необходимо:

- одноразовое офлайн уведомление. Используется если необходимо совершить одиночный безналичный или наличный расчет. Пользователь может создать заголовок, текст и время события;

- одноразовое онлайн уведомление. Используется если необходимо совершить одиночный безналичный или наличный расчет за услугу, которую поддерживает система ЕРИП (Единое Расчётное и Информационное Пространство). Пользователь может ввести реквизиты необходимые для оплаты данной услуги (например, телефонный номер, личный идентификатор), сумму для оплаты и периодичность (по умолчанию 30 дней). В данном случае при уведомлении пользователя ему будет предложено оплатить услугу

в пару касаний экрана;

- периодическое напоминание о платеже. Используется если необходимо напоминание о скорой необходимости платежа за периодическую услугу (абонентская плата за интернет, телефон). В случае если оплата данной услуги возможна в системе ЕРИП, пользователь может ввести реквизиты и сумму для оплаты;

- периодическое напоминание о платеже по календарному графику. Используется в случае, когда способ выше не подходит т.к. не важно количество прошедших суток, а необходима оплата до конкретной даты месяца. Пользователь может выбрать заголовок, описание услуги, реквизиты ЕРИП (если возможно и необходимо) и дату до которой необходимо совершить платеж. Приложение автоматически будет предупреждать пользователя за несколько дней о приближающемся платеже и предлагать совершить оплату в пару кликов;

- собственный график события. Используется если пользователю необходимо установить данный платеж на определенные дни. При создании события он может создать заголовок, описание, сумму к оплате, реквизиты ЕРИП (если возможно и необходимо), после чего на календаре сможет отметить нужное количество дней на следующий год.

Пользователь может в любой момент удалять события, редактировать, либо ставить их на паузу на необходимое количество дней, либо до определенной даты.

Система ЕРИП имеет возможность выставления счета на некоторые услуги (оплата учебы, некоторых услуг ЖКХ и так далее), следовательно, зная идентификатор пользователя в данной услуге, приложение может само получить необходимую сумму к оплате. При создании событий пользователь может указать поведение программы в данном случае: предлагать полученную сумму, не оплачивать в случае если суммы нет или игнорировать данные ЕРИП и оплатить сумму, указанную пользователем.

Так как приложение работает с реальными деньгами пользователя, каждое его одобрение оплаты должно быть подтверждено введением личного пароля. Пользователь может установить пароль при регистрации, которая запускается при первом запуске программы. Уникальным ключом каждого пользователя является номер мобильного телефона, изменить его можно в настройках программы. Пользователь может подключать к программе несколько платежных карт различных банков или платежных систем, либо другие способы оплаты (интернет-кошельки, оплата со счета мобильного) и в будущем, при создании событий, сможет выбрать с помощью какого способа оплаты оформить услугу. Также возможно выбирать способ оплаты перед каждым непосредственным платежом.

Работа с платежными системами и системой расчета ЕРИП накладывает на приложение большую ответственность за информационную безопасность данных пользователя и всех проводимых операций. Необходима шифровка данных при любых синхронизациях с сервером приложения и наличие безопасного соединения при отправке команд в систему расчета.

Список использованных источников:

1. ЕРИП Расчет - Платежные агенты [Электронный ресурс]. – Режим доступа: <http://raschet.by/bankam/platezhnye-agenty/>. Дата доступа 23.04.2017.

ЦЕЛЕОРИЕНТИРОВАННОЕ МОДЕЛИРОВАНИЕ И ПРОЕКТИРОВАНИЕ

Институт информационных технологий БГУИР, г. Минск, Республика Беларусь

Биркос В.Д.

Савенко А.Г. – магистр технических наук, ассистент

С момента начала эпохи информационных технологий, процесс разработки программного обеспечения претерпел серьезные изменения. Теперь в процессе принимают участие не только программисты, непосредственно занимающиеся созданием программного продукта, но и тестировщики, менеджеры, дизайнеры. Делегирование задач обеспечивает создание более качественного программного средства, за счет участия в процессе разработки узкоспециализированных специалистов. Все их усилия будут направлены на реализацию продукта, который будет не только удовлетворять всем требованиям пользователя, но и станет желанным. Для решения такой задачи было разработано целеориентированное проектирование.

Целеориентированное проектирование - это подход к проектированию пользовательского интерфейса, основанный на персонажах и их целях [2]. Процесс принято разделять на шесть стадий [1]: сопровождение, детализация, формирование общей инфраструктуры, создания требований, моделирования и исследования.

1) Первая стадия - исследование

На данной стадии может использоваться ряд методов для исследования пользователей: интервью, конкурентный анализ, маркетинговые исследования. Результатом будет являться набор характерных поведенческих моделей (поведенческих шаблонов). Их дальнейший анализ позволит определять цели и мотивы конечных пользователей.

2) Вторая стадия - моделирование

На стадии моделирования, формируется две модели: предметной области и пользователей. В качестве основы берутся шаблоны рабочих процессов и поведенческих шаблонов. Пользовательские модели представляют собой образец поведения пользователей, которые представляют собой различные устойчивые комбинации целей, взглядов и мотивов [1]. Модели предметной состоят из диаграммы рабочих процессов.

3) Третья стадия – создание требований