

- использование мобильного устройства в качестве манипулятора ввода персонального компьютера;
- перемещение курсора;
- осуществление нажатий клавиш мыши;
- набор текста с помощью виртуальной клавиатуры;
- управление медиа возможностями персонального компьютера:
- воспроизведение;
- остановка;
- переключение композиций;
- изменение уровня громкости;
- переключение слайдов в офисном программном обеспечении;
- приглушение звука при поступлении входящего или начале исходящего звонка на мобильном устройстве;
- получение информации о поступивших уведомлениях на мобильное устройство и ответ на них, если он подразумевается;
- информация об уровне заряда мобильного устройства;
- приём и передача файлов с персонального компьютера на мобильное устройство и наоборот;

Исходя из вышесказанного, стоит отличать данное программное обеспечение дистанционного управления от систем удалённого доступа, таких как программа TeamViewer или протокол передачи VNC. Данное мобильное приложение подразумевает тесную интеграцию полноразмерных и мобильных устройств в единый комплекс, нежели удалённый доступ с дублированием экрана.

Данное мобильное приложение поставляется в паре с серверным приложением для персонального компьютера для обеспечения работы протокола клиент-сервер между устройствами. Серверная часть разрабатывается с использованием языка программирования C# на платформе .NETCore, предоставляемой и поддерживаемой Microsoft. Эта современная платформа позволяет в будущем переиспользовать большую часть кода для переноса серверного кода на GNU/Linux и, возможно, для более закрытой с точки зрения разработчика программного обеспечения macOS.

На операционной системе Microsoft Windows для управления возможностями персонального компьютера используется функционал, предоставляемый подсистемой Win32 (WINAPI), состоящей из неуправляемого кода.

Клиентская часть реализована для платформы Android, самой популярной мобильной платформы в мире, с использованием языка программирования Kotlin и языка разметки XML. Обмен данными между устройствами происходит с помощью локальной сети по протоколу UDP. С UDP компьютерные приложения могут посылать сообщения (в данном случае называемые датаграммами) другим хостам по IP-сети без необходимости предварительного сообщения для установки специальных каналов передачи или путей данных, что позволяет ускорить обработку поступающей информации. В дальнейшем существует возможность использовать помимо локальной сети прямое подключение при помощи протоколов Wi-FiDirect и Bluetooth, что позволит использовать данную систему автономно. Протокол взаимодействия между клиентом и сервером будет стандартизирован, документирован и открыт для доработки, что позволит для каждой индивидуальной платформы реализовать свой независимый функционал.

Уникальность системы заключается в возможности использовать её во многих направлениях. Например, превратить в любой персональный компьютер в локальной сети в медиацентр, подключив его, например, к большому экрану. Или же использовать экран мобильного устройства в качестве графического планшета.

На данный момент реализована функция манипуляции курсором и нажатия, но изучена возможность реализации всего остального функционала на платформах Microsoft Windows и GNU/Linux и будет производиться доработка и дальнейшее развитие как самого протокола связи, так и их реализаций для конкретных платформ.

Таким образом, данный программный комплекс поможет более тесно интегрировать устройства между собой в единую систему для более удобной и эффективной работы с каждым из них.

СЕРВИС ДЛЯ КОНТРОЛЯ ВЫПОЛНЕНИЯ ЗАДАЧ

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

Неволько Н.Д.

Жвакина А.В. – к.т.н., доцент

В современном мире информационных технологий, при работе над новым проектом с использованием гибких методологий разработки программного обеспечения, нередко встает вопрос: как организовать эффективное разделение и отслеживание выполнения задач между различными группами специалистов (программистами, тестировщиками, бизнес-аналитиками и др.)? Разработанный сервис для контроля выполнения задач предназначен для решения данной задачи.

При использовании Agile методологий, как например Scrum или Kanban, задачи делятся на подзадачи

и выносятся на зонированную общую доску. Но как быть, если члены команды работают удаленно, да еще и в разных часовых поясах? А как управлять проектом и задачами, если команда состоит из более чем ста человек? Здесь на помощь придет решение в виде веб-приложения по управлению задачами проекта.



Рис. 1 – Пример Agile доски с задачами

Применение данного приложения поможет руководителям организовать работу в командах, создавать свой рабочий процесс для каждого вида задач, а интуитивно понятный интерфейс позволяет представить статистику и текущие статусы выполнения задач в удобном виде. Система обеспечивает доступ в любое время и в любом месте. Благодаря выбранному современному стеку технологий данные пользователей надежно защищены.

В основе принципа работы лежат задачи (Task, Ticket) – атомарные единицы работы над проектом, которые по мере своего выполнения изменяют свой текущий статус, двигаясь по рабочему процессу. Основной функционал включает в себя:

- создание задач и подзадач;
- организацию своего рабочего процесса для каждого вида задач;
- возможность учета запланированного и фактически затраченного времени;
- комментирование задач с целью лучшей коммуникации между исполнителями задач.

Достоинствами предоставляемого сервиса являются:

- отказ от представления задач и подзадач на старомодных «физических» досках, перенос данных на сервер, доступный круглосуточно;
- удобный и понятный интерфейс;
- возможность использования любыми рабочими группами в любых проектах;
- учет затраченного времени над задачами каждого сотрудника;
- наглядность текущего статуса исполнения всех задач на общей доске задач;
- безопасность данных пользователей;
- использование современных технологий в разработке.

Итогом работы является полноценное веб-приложение, которое поможет оптимизировать контроль выполнения задач, улучшить коммуникацию между сотрудниками, а значит, и увеличить скорость и качество создания программных продуктов.

Список использованных источников:

1. Гибкая методология разработки. [Электронный ресурс] – Режим доступа: https://ru.wikipedia.org/wiki/Гибкая_методология_разработки. – Дата доступа : 08.04.2018.

БАЗИСЫ ГРЁБНЕРА. АЛГОРИТМ БУХБЕРГЕРА

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

Новак Д.С.

Калугина М.А.–канд. физ.-мат. наук, доцент

В течение всей истории математики стоял вопрос о решении систем алгебраических уравнений с большими целочисленными коэффициентами. Австрийский математик Вольфганг Грёбнер еще в 1930-х годах разрабатывал теорию стандартных базисов, которая позволяла разрешать различные САУ, однако его работа не нашла практического применения из-за больших практических подсчетов, которые занимали много времени, а иногда были вовсе неразрешимы.

В начале 60-х годов прошлого века был достигнут прогресс в области систем алгебраических уравнений. Австрийский математик Бруно Бухбергер, ученик Грёбнера ввел такие понятия как базис Грёбнера, а также разработал эффективный алгоритм, позволяющий находить базис Грёбнера для систем