

РАЗРАБОТКА ОФЛАЙН-КОМПИЛЯТОРА ЯЗЫКА PASCAL НА ANDROID

Введение. Pascal — один из наиболее известных языков программирования, используется для обучения программированию в старших классах и на первых курсах вузов, является основой для ряда других языков.

Разработка Android-приложения для компиляции программного кода на языке Pascal позволит любому человеку при наличии Android-устройства без интернета проверить работоспособность кода или же просто написать свою программу.

Основная часть. Приложение является компилятором языка Pascal на Android. Это приложение создано для всех, чтобы запускать Pascal-код на мобильном устройстве без компьютера, так программист сможет проверить работоспособность своего кода либо же написать свой код в любом месте при наличии Android-устройства. Для разработки приложения использовалась среда разработки Android Studio и язык программирования Java.

Android Studio — это интегрированная среда разработки (IDE) для работы с платформой Android, выпущенная компанией Google. Android Studio совместима с платформой Google App Engine для быстрой интеграции в облаке новых API и функций. В среде разработки вы найдете различные API, такие как Google Play, Android Pay и Health. Присутствует поддержка всех платформ Android, начиная с Android 4.6 [10].

Главное окно программы представлено на рисунке 1. При запуске программы открывается окно с результатом компиляции и сообщение об успешном выполнении (рисунок 2). Также компилятор содержит в себе встроенный отладчик кода. Пример отладки представлен на рисунке 3.

Основные особенности офлайн-компилятора: 1) компиляция программ на языке Pascal и запуск их без интернета; 2) сообщение об ошибке при компиляции; 3) мощный редактор с множеством интеллектуальных функций: меню Файл: создать новый программный файл, открыть, сохранить автоматическое сохранение файла; меню редактирования: Отмена, повтор, копировать, вставить; авто-подсказки: отображение небольшого всплывающего окна, которое предлагает слово, совпадающую с набираемым словом; автоформат: автоматическое форматирование кода для более удобного просмотра; найти / найти и заменить: поддержка регулярных выражений; перемещение курсора на строку; выделение кода: выделение ключевых слов определенным цветом; стиль кода: модный интерфейс редактора; размер шрифта, шрифт, перенос слов; 4) Поддержка библиотеки Android (датчик, батарея, запись звука, камера и др.): библиотека «aTTSpeech» преобразует текст в речь; библиотека «aRecognition» преобразует речь в текст (требуется Google Voice); библиотека «aVibrate» поддерживает контроль вибрации устройства; библиотека «aSensor» поддерживает обработку датчиков (света, акселерометра и др.); библиотека «aNotify» поддерживает отображение уведомлений в строке состояния; библиотека «aClipboard» работает с буфером обмена на Android; библиотека «aBattery» извлекает информацию о состоянии батареи устройства.

Рисунок 1 — Главное окно программы

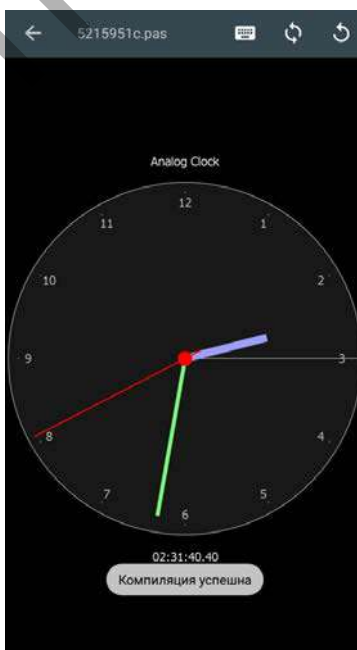


Рисунок 2 — Результат компиляции

Рисунок 3 — Пример отладки программы

Заключение. Разработан офлайн-компилятор языка Pascal для устройств на операционной системе Android, которые полностью поддерживает все возможности компилятора языка, как и на персональном компьютере, а также множество дополнительных библиотек для Android-устройств.

Список цитируемых источников

1. Android Studio [Электронный ресурс]. — Режим доступа: <http://wnfx.ru/android-studio-ide-ot-google/>. — Дата доступа: 18.02.2019.

УДК 004.421

А. А. Ермакова

учреждение образования «Барановичский государственный университет», Барановичи

ВЕБ-ПРИЛОЖЕНИЕ «ЕЖЕДНЕВНИК»

Введение. В современном быстроразвивающемся мире повышается занятость человека: множество обязанностей, целей и задач, которые, зачастую, нужно реализовать в течение всего лишь одного дня. Венцом успеха будет являться правильное распределение времени.

Основная часть. В правильном распределении времени помогает ежедневник: человек грамотно расписывает свой день буквально по минутам и, как итог, успевает всегда и везде. На данный момент актуальность использования web-приложения «Ежедневник» очень высока, так как ежедневник помогает не упускать из виду ни одной детали, человек может анализировать эффективность своей работы и даже находить ошибки в своих поступках или действиях.

Целью работы является разработка web-приложения «Ежедневник».

Достижение поставленной цели требует решения следующих задач: разработка методов и моделей представления ежедневника; разработка информационной модели подсистемы (структуры базы данных) учета пользователей и наполнение разработанной БД соответствующей информацией о записях; тестирование программы с использованием разработанной БД; описание алгоритмов программных модулей; описание полученных результатов.

Веб-приложение — клиент-серверное приложение, в котором клиент взаимодействует с сервером при помощи браузера, а за сервер отвечает веб-сервер. Логика веб-приложения распределена между сервером и клиентом, хранение данных осуществляется, преимущественно, на сервере, обмен информацией происходит по сети. Одним из преимуществ такого подхода является тот факт, что клиенты не зависят от конкретной операционной системы пользователя, поэтому веб-приложения являются межплатформенными службами [1].

Клиент-сервер — вычислительная или сетевая архитектура, в которой задания или сетевая нагрузка распределены между поставщиками услуг (сервисов), называемыми серверами, и заказчиками услуг, называемыми клиентами.

Сервер — это программа, представляющая какие-то услуги другим программам и обслуживающая запросы клиентов на получение ресурсов определенного вида. Клиент — это программа, использующая услугу, представляемую программой сервера.

Часто люди клиентом или сервером просто называют компьютер, на котором работает какая-либо из этих программа. В сущности, клиент и сервер — это роли, исполняемые программами. Клиенты и сервера физически могут находиться на одном компьютере. Одна и та же программа может быть и клиентом, и сервером одновременно [2].

Java-WEB приложение представляет собой систему, которая реализует структуру MVC(Model-view-controller). В данной структуре за отображение информации отвечает «view» а за реакции на действия пользователя «controller».

Приложение состоит из двух составляющих:

— сервера (в данной работе используется Tomcat 9);

— клиента (может использоваться любой web-браузер, в данном проекте использован Google Chrome).

Окно клиента представлено на рисунке 1.

Для того, чтобы войти в аккаунт либо зарегистрировать новый, необходимо пройти на страницу «авторизация/регистрация». Страница с формами «авторизация/регистрация» представлена на рисунке 2.

Личная страница пользователя представлена на рисунке 3.