**Министерство образования Республики Беларусь**

**Белорусский государственный университет**

**МЕХАНИКО-МАТЕМАТИЧЕСКИЙ ФАКУЛЬТЕТ**

**Кафедра теории функций**

**Отчёт**

**о прохождении преддипломной практики**

**на предприятии**

**ООО «АйБиЭй IT-парк»**

**Выполнил**: Студент 4 курса

Группы №6

Киселев О.А

**Руководитель от ВУЗа:** Старший преподаватель

Долгополова Ольга Борисовна

**Руководитель от предприятия:** Заместитель директора

Милешко Александра Вячеславовна

**Минск 2019**

**Оглавление**

[Введение 3](#_Toc419756548)

1. Используемые операционные системы [4](#_Toc419756558)  
[1.1. Andoid 4](#_Toc419756559)  
[1.2. iOS 5](#_Toc419756559)

2. [Используемые технологии 6](#_Toc419756563)

[2.1. Cordova 6](#_Toc419756564)  
[2.2. TypeScript 7](#_Toc419756565)  
[2.3. Angular 2 8](#_Toc419756559)  
[2.4. Ionic 2 9](#_Toc419756559)  
[2.5. SASS 10](#_Toc419756559)  
[2.6. HTML 5 11](#_Toc419756559)

[Заключение 13](#_Toc419756566)

Введение  
Благодаря техническому прогрессу, ежедневно появляется масса новейших разработок в различных технических областях. В настоящее время разрабатывается огромное количество различных приложений и сайтов, упрощающих жизнь среднестатистического пользователя: прогноз погоды, фоторедакторы, мессенджеры, переводчики, карты и т.д. Отдельную нишу занимают образовательные приложения, например, изучение иностранного языка или карманный помощник по какому-нибудь школьному предмету. Талантливые и способные студенты могут создавать приложения, помогающие им в учебе, например, вместо того, чтобы вручную считать производную от громоздкой функции, можно просто ввести ее в поле приложения. Иными словами, приложения заменяют трудоемкий и долгий путь на быстрый и эффективный.

Цели производственной практики:

* Закрепить и углубить знания, полученные во время обучения в университете
* Приобрести практические навыки работы

Были поставлены следующие задачи:

* Создать концепт и разработать дизайн будущего приложения
* Изучить и проанализировать экономический эффект будущего приложения
* Подробнее узнать о технологиях Cordova, SASS, HTML5, TypeScript, Angular 2, Ionic 2
* Выполнить индивидуальное задание: создание приложения Calculus

1. Используемые операционные системы

1.1 Android

Андроид (от англ. Android) — это название программной платформы для мобильных устройств, разработанная компанией Google. Эта платформа включает в себя операционную систему базирующуюся на ядре Linux 2.6, связующее ПО и ключевые приложения. Распространяется она под [лицензией Apache 2.0.](http://apache.org/licenses/)

Основные элементы Android:   
Application framework – обеспечивает возможность гибкой работы с компонентами системы, используя API (Application Program Interface);

* Dalvik virtual machine – оптимизированная для мобильных устройств виртуальная машина
* Integrated browser – встроенный интернет браузер базирующийся на открытом движке WebKit
* Optimized graphics – поддержка пользовательских 2D графических библиотек; для реализации 3D графики используется OpenGL ES 1.0 (при этом аппаратное ускорение необязательно)
* SQLite – используется для хранения структурированных данных (аналогично и в iPhone)
* Media support – поддержка аудио, видео и графических форматов (MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF)
* GSM Telephony, Bluetooth, EDGE, 3G, and WiFi, Camera, GPS, compass, and accelerometer – зависит от аппаратного обеспечения конкретного мобильного устройства
* Rich development environment – включает в себя эмулятор, инструменты для отладки, профилирования памяти и производительности (Profiler), а также плагин для Eclipse IDE

1.2 iOS

**iOS**- операционная система корпорации Apple, устанавливаемая ею на мобильные телефоны iPhone 3-X, планшеты iPad, а также на музыкальный плеер iPod.  
Достоинства iOS:

1. Самый крупный магазин приложений с достаточно качественными приложениям
2. Быстрота работы системы по сравнению с другими
3. Хорошее качество телефонов и планшетов компании Apple
4. Быстрая реакция на ошибки  и отсутствие вирусов
5. Красота интерфейса и графики.
6. Постоянное обновление системы раз в год, в т.ч. и для старых устройств

2. Используемые технологии

2.1 Cordova

**Apache Cordova** — мобильная среда разработки приложений, первоначально разработанная [Nitobi](https://ru.wikipedia.org/w/index.php?title=Nitobi&action=edit&redlink=1" \o "Nitobi (страница отсутствует)). Компания [Adobe Systems](https://ru.wikipedia.org/wiki/Adobe_Systems" \o "Adobe Systems) приобрела Nitobi в 2011 году и провела [ребрендинг](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%B1%D1%80%D0%B5%D0%BD%D0%B4%D0%B8%D0%BD%D0%B3" \o "Ребрендинг) [PhoneGap](https://ru.wikipedia.org/wiki/PhoneGap" \o "PhoneGap), после чего выпустила версию с открытым исходным кодом программного обеспечения под названием Apache Cordova[[2]](https://ru.wikipedia.org/wiki/Cordova" \l "cite_note-2). Apache Cordova позволяет программистам создавать приложения для мобильных устройств с помощью [CSS3](https://ru.wikipedia.org/wiki/CSS3), [HTML5](https://ru.wikipedia.org/wiki/HTML5) и [JavaScript](https://ru.wikipedia.org/wiki/JavaScript" \o "JavaScript), вместо того, чтобы использовать конкретные платформы [API](https://ru.wikipedia.org/wiki/API), такие как [Android](https://ru.wikipedia.org/wiki/Android" \o "Android), [IOS](https://ru.wikipedia.org/wiki/IOS) или [Windows Phone](https://ru.wikipedia.org/wiki/Windows_Phone" \o "Windows Phone)[[3]](https://ru.wikipedia.org/wiki/Cordova#cite_note-3). Это обеспечивается за счет преобразования из CSS, HTML и JavaScript в код, который любая платформа воспринимает как элемент web. Это расширяет возможности HTML и JavaScript для работы с различными устройствами. В результате приложения являются гибридными, это означает, что они не являются ни по-настоящему мобильными приложениями (потому что вся генерация макета осуществляются с помощью [web-view](https://ru.wikipedia.org/w/index.php?title=Web-view&action=edit&redlink=1" \o "Web-view (страница отсутствует))вместо основной структуры пользовательского интерфейса платформы), ни web — потому как они не только web — приложение, но и упакованы в качестве приложения для распределения, а также имеет доступ к API базового функционала устройства, такого как файловая система, камера и пр.). [ПО](https://ru.wikipedia.org/wiki/%D0%9F%D0%9E) с открытым исходным кодом, Apache Cordova используется в других программах, таких как [Appery.io](https://ru.wikipedia.org/w/index.php?title=Appery.io&action=edit&redlink=1) или [Intel XDK](https://ru.wikipedia.org/wiki/Intel_XDK" \o "Intel XDK).

2.2 TypeScript

**TypeScript** — [язык программирования](https://ru.wikipedia.org/wiki/%D0%AF%D0%B7%D1%8B%D0%BA_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F), представленный [Microsoft](https://ru.wikipedia.org/wiki/Microsoft" \o "Microsoft) в [2012 году](https://ru.wikipedia.org/wiki/2012_%D0%B3%D0%BE%D0%B4) и позиционируемый как средство разработки [веб-приложений](https://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D0%BF%D1%80%D0%B8%D0%BB%D0%BE%D0%B6%D0%B5%D0%BD%D0%B8%D0%B5), расширяющее возможности [JavaScript](https://ru.wikipedia.org/wiki/JavaScript" \o "JavaScript).

Разработчиком языка TypeScript является [Андерс Хейлсберг](https://ru.wikipedia.org/wiki/%D0%A5%D0%B5%D0%B9%D0%BB%D1%81%D0%B1%D0%B5%D1%80%D0%B3,_%D0%90%D0%BD%D0%B4%D0%B5%D1%80%D1%81) ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *Anders Hejlsberg*), создавший ранее [Turbo Pascal](https://ru.wikipedia.org/wiki/Turbo_Pascal" \o "Turbo Pascal), [Delphi](https://ru.wikipedia.org/wiki/Delphi_(%D1%8F%D0%B7%D1%8B%D0%BA_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F)" \o "Delphi (язык программирования)) и [C#](https://ru.wikipedia.org/wiki/C_Sharp).

TypeScript является обратно совместимым с JavaScript и компилируется в последний. Фактически, после компиляции программу на TypeScript можно выполнять в любом современном браузере или использовать совместно с серверной платформой [Node.js](https://ru.wikipedia.org/wiki/Node.js). Код экспериментального компилятора, транслирующего TypeScript в JavaScript, распространяется под [лицензией Apache](https://ru.wikipedia.org/wiki/%D0%9B%D0%B8%D1%86%D0%B5%D0%BD%D0%B7%D0%B8%D1%8F_Apache). Его разработка ведётся в публичном репозитории через сервис [GitHub](https://ru.wikipedia.org/wiki/GitHub" \o "GitHub)[[8]](https://ru.wikipedia.org/wiki/TypeScript#cite_note-8).

TypeScript отличается от JavaScript возможностью [явного](https://ru.wikipedia.org/w/index.php?title=%D0%AF%D0%B2%D0%BD%D0%B0%D1%8F_%D1%82%D0%B8%D0%BF%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F&action=edit&redlink=1)[[en]](https://en.wikipedia.org/wiki/Manifest_typing) [статического](https://ru.wikipedia.org/wiki/%D0%A1%D1%82%D0%B0%D1%82%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B0%D1%8F_%D1%82%D0%B8%D0%BF%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F) назначения типов, поддержкой использования полноценных классов (как в традиционных объектно-ориентированных языках), а также поддержкой подключения модулей, что призвано повысить скорость разработки, облегчить читаемость, [рефакторинг](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D1%84%D0%B0%D0%BA%D1%82%D0%BE%D1%80%D0%B8%D0%BD%D0%B3" \o "Рефакторинг) и повторное использования кода, помочь осуществлять поиск ошибок на этапе разработки и компиляции, и, возможно, ускорить выполнение программ.

Планируется, что в силу полной обратной совместимости адаптация существующих приложений на новый язык программирования может происходить поэтапно, путём постепенного определения типов.

На момент релиза представлены файлы для восприятия расширенного синтаксиса TypeScript для [Vim](https://ru.wikipedia.org/wiki/Vim" \o "Vim) и [Emacs](https://ru.wikipedia.org/wiki/Emacs" \o "Emacs), а также плагин для [Microsoft Visual Studio](https://ru.wikipedia.org/wiki/Microsoft_Visual_Studio" \o "Microsoft Visual Studio).

Одновременно с выходом спецификации разработчики подготовили файлы с декларациями статических типов для некоторых популярных [JavaScript-библиотек](https://ru.wikipedia.org/wiki/%D0%91%D0%B8%D0%B1%D0%BB%D0%B8%D0%BE%D1%82%D0%B5%D0%BA%D0%B0_JavaScript" \o "Библиотека JavaScript), среди которых [jQuery](https://ru.wikipedia.org/wiki/JQuery" \o "JQuery).

2.3 Angular 2

**AngularJS** — [JavaScript](https://ru.wikipedia.org/wiki/JavaScript" \o "JavaScript)-[фреймворк](https://ru.wikipedia.org/wiki/%D0%A4%D1%80%D0%B5%D0%B9%D0%BC%D0%B2%D0%BE%D1%80%D0%BA) с [открытым исходным кодом](https://ru.wikipedia.org/wiki/%D0%9E%D1%82%D0%BA%D1%80%D1%8B%D1%82%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B5_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D0%B5). Предназначен для разработки [одностраничных приложений](https://ru.wikipedia.org/wiki/%D0%9E%D0%B4%D0%BD%D0%BE%D1%81%D1%82%D1%80%D0%B0%D0%BD%D0%B8%D1%87%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%B8%D0%BB%D0%BE%D0%B6%D0%B5%D0%BD%D0%B8%D0%B5)[[4]](https://ru.wikipedia.org/wiki/AngularJS#cite_note-4). Его цель — расширение браузерных приложений на основе [MVC](https://ru.wikipedia.org/wiki/Model-View-Controller)-шаблона, а также упрощение тестирования и разработки.

Фреймворк работает с [HTML](https://ru.wikipedia.org/wiki/HTML), содержащим дополнительные пользовательские атрибуты, которые описываются директивами, и связывает ввод или вывод области страницы с моделью, представляющей собой обычные переменные JavaScript. Значения этих переменных задаются вручную или извлекаются из статических или динамических JSON-данных.

2.4 Ionic 2

Ionic - это полноценный SDK с открытым исходным кодом для разработки мобильных приложений. Оригинальная версия была выпущена в 2013 году и построена на основе AngularJS и Apache Cordova. Более поздние выпуски, известные как Ionic 2 или просто «Ionic», построены на Angular. Ionic предоставляет инструменты и услуги для разработки гибридных мобильных приложений с использованием веб-технологий, таких как CSS, HTML5 и Sass. Приложения можно создавать с помощью этих веб-технологий, а затем распространять через собственные магазины приложений, которые будут установлены на устройствах, используя Cordova. Ионик был создан Максом Линчем, Бен Сперри и Адамом Брэдли из Drifty Co. в 2013 году. Ionic Creator - инструмент для создания интерфейса с помощью мыши.

2.5 SASS

Sass (Syntactically Awesome StyleSheets)

Sass - это расширение CSS, которое добавляет силу и элегантность базовому языку. Он позволяет использовать переменные, вложенные правила, встроенный импорт и многое другое, все полностью совместимо с CSS синтаксисом. Sass помогает сохранять большие таблицы стилей хорошо организованными и быстро запускать небольшие стили, особенно с помощью библиотеки стилей Compass.

Особенности:

* Полностью совместимый с CSS
* Расширения языка, такие как переменные, вложенность и mixins
* Многие полезные функции для управления цветами и другими значениями
* Расширенные функции, такие как директивы управления для библиотек
* Хорошо отформатированный, настраиваемый вывод

2.6 HTML5

**HTML5** ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *HyperText Markup Language, version 5*) — [язык для структурирования и представления](https://ru.wikipedia.org/wiki/%D0%AF%D0%B7%D1%8B%D0%BA_%D1%80%D0%B0%D0%B7%D0%BC%D0%B5%D1%82%D0%BA%D0%B8) содержимого [всемирной паутины](https://ru.wikipedia.org/wiki/%D0%92%D1%81%D0%B5%D0%BC%D0%B8%D1%80%D0%BD%D0%B0%D1%8F_%D0%BF%D0%B0%D1%83%D1%82%D0%B8%D0%BD%D0%B0). Это пятая версия [HTML](https://ru.wikipedia.org/wiki/HTML). Хотя стандарт был завершён (рекомендованная версия к использованию) только в 2014 году[[1]](https://ru.wikipedia.org/wiki/HTML5#cite_note-1)[[2]](https://ru.wikipedia.org/wiki/HTML5#cite_note-w3c-2) (предыдущая, четвёртая, версия опубликована в 1999 году[[3]](https://ru.wikipedia.org/wiki/HTML5#cite_note-3)), ещё с 2013 года[[4]](https://ru.wikipedia.org/wiki/HTML5#cite_note-HTML5-20131029-4) браузерами оперативно осуществлялась поддержка, а разработчиками — использование рабочего стандарта ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *HTML Living Standard*). Цель разработки HTML5 — улучшение уровня поддержки [мультимедиа](https://ru.wikipedia.org/wiki/%D0%9C%D1%83%D0%BB%D1%8C%D1%82%D0%B8%D0%BC%D0%B5%D0%B4%D0%B8%D0%B0)-технологий с одновременным сохранением обратной совместимости, удобочитаемости кода для человека и простоты анализа для [парсеров](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D0%BD%D1%82%D0%B0%D0%BA%D1%81%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B8%D0%B9_%D0%B0%D0%BD%D0%B0%D0%BB%D0%B8%D0%B7" \o "Синтаксический анализ).

Во всемирной паутине долгое время использовались стандарты [HTML](https://ru.wikipedia.org/wiki/HTML) 4.01, [XHTML](https://ru.wikipedia.org/wiki/XHTML) 1.0 и XHTML 1.1. [Веб-страницы](https://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D1%81%D1%82%D1%80%D0%B0%D0%BD%D0%B8%D1%86%D0%B0) на практике оказывались свёрстаны с использованием смеси особенностей, представленных различными спецификациями, включая спецификации программных продуктов, например [веб-браузеров](https://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D0%B1%D1%80%D0%B0%D1%83%D0%B7%D0%B5%D1%80), а также сложившихся общеупотребительных приёмов. HTML5 был создан как единый [язык разметки](https://ru.wikipedia.org/wiki/%D0%AF%D0%B7%D1%8B%D0%BA_%D1%80%D0%B0%D0%B7%D0%BC%D0%B5%D1%82%D0%BA%D0%B8), который мог бы сочетать [синтаксические нормы](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D0%BD%D1%82%D0%B0%D0%BA%D1%81%D0%B8%D1%81_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)) HTML и XHTML. Он расширяет, улучшает и рационализирует разметку документов, а также добавляет единый [API](https://ru.wikipedia.org/wiki/API) для сложных [веб-приложений](https://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D0%BF%D1%80%D0%B8%D0%BB%D0%BE%D0%B6%D0%B5%D0%BD%D0%B8%D0%B5)[[5]](https://ru.wikipedia.org/wiki/HTML5#cite_note-HTML5diffHTML4-5).

В HTML5 реализовано множество новых синтаксических особенностей. Например, элементы <video>, <audio> и <canvas>, а также возможность использования [SVG](https://ru.wikipedia.org/wiki/SVG) и [математических формул](https://ru.wikipedia.org/wiki/MathML). Эти новшества разработаны для упрощения создания и управления графическими и мультимедийными объектами в сети без необходимости использования сторонних [API](https://ru.wikipedia.org/wiki/API) и [плагинов](https://ru.wikipedia.org/wiki/%D0%9F%D0%BB%D0%B0%D0%B3%D0%B8%D0%BD). Другие новые элементы, такие как <section>, <article>, <header> и <nav>, разработаны для того, чтобы обогащать [семантическое](https://ru.wikipedia.org/wiki/%D0%A1%D0%B5%D0%BC%D0%B0%D0%BD%D1%82%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B0%D1%8F_%D0%B2%D1%91%D1%80%D1%81%D1%82%D0%BA%D0%B0) содержимое документа (страницы). Новые атрибуты были введены с той же целью, хотя ряд элементов и атрибутов был удалён. Некоторые элементы, например <a>, <menu> и <cite>, были изменены, переопределены или стандартизированы. API и [DOM](https://ru.wikipedia.org/wiki/Document_Object_Model) стали основными частями спецификации HTML5[[5]](https://ru.wikipedia.org/wiki/HTML5#cite_note-HTML5diffHTML4-5). HTML5 также определяет некоторые особенности обработки ошибок вёрстки, поэтому синтаксические ошибки должны рассматриваться одинаково всеми совместимыми браузерами[[6]](https://ru.wikipedia.org/wiki/HTML5#cite_note-6).

Заключение

За время производственной практики мы научились использовать следующие технологии: Angular 2, Cordova, HTML5, SASS, Ionic 2, TypeScript. Улучшили свои навыки в Adobe Photoshop. Изучили новые хитрости верстки. Поучаствовали в создании приложения Calculus, которое предназначено для студентов. Это приложение позволяет быстро вычислить сложные формулы, посчитать производную, интегралы, решить уравнения с несколькими переменными, а также в нём есть документация, которой можно воспользоваться без возможности подключения к интернету. В приложении есть сборник лекций от известного европейского математика.