# ВВЕДЕНИЕ

В качестве моего дипломного проекта разрабатывается автоматизированная система учета посещаемости и успеваемости студентов. Система предусматривает клиентскую и серверную часть. Для клиентской части АС планируется разработать мобильное приложение для ОС Android, которое будет предоставлять возможность работы с электронными ведомостями. Планируется реализовать ведомости для занятий лекционного типа и занятий для выполнения лабораторных работ.

Разрабатываемая АС будет решать практически жизненно важные задачи – вести учет посещаемости и успеваемости студентов. Это одни из важнейших показателей, определяющие обучение студента в ВУЗе. Поэтому важно спроектировать качественный интерфейс для мобильного приложения.

В данной работе будет рассмотрено проектирование интерфейса мобильного приложения для ведомостей лекционных занятий.

# АНАЛИТИЧЕСКАЯ ЧАСТЬ

Система разрабатывается для преподавателей. Делается мобильное приложение под Android. Важно сделать хороший интерфейс.

## Анализ работы с ведомостями занятий

Для формирования требований к интерфейсу необходимо понять, как происходит работа с ведомостями на данный момент.

Рассмотрим использование ведомостей во время лекционного занятия. На лекционных занятиях обычно бывает более 1 группы. Отметим основные варианты использования и действия, совершаемые при этом преподавателем:

### перед занятием преподаватель должен найти ведомости всех групп, для которых проводится занятие:

### восстановить в памяти названия необходимых групп (например, вспомнить);

### среди всех ведомостей найти нужные ведомости;

### спустя некоторое время после начала занятия, преподаватель проверяет посещаемость студентов всех групп (для всех ведомостей):

### взять ведомость очередной группы;

### проверить посещаемость студентов группы (для всех студентов):

### спросить в аудитории студента;

### в зависимости от ответа отметить посещаемость;

### после проверки посещаемости группы преподаватель может обратить внимание на уровень посещаемости, например, присутствует мало студентов;

### в течение занятия преподаватель может оставить заметку на полях ведомости на следующие занятия;

### в течение занятия могут приходить опоздавшие студенты, которых нужно отмечать:

### найти ведомость с нужной группой;

### найти в списке нужного студента;

### найти в столбце с посещаемостью нужную ячейку;

### отметить опоздание.

Рассмотрим, где преподаватель может наиболее вероятно совершить ошибку при работе с ведомостью. Стоит отметить, что чтение лекции – сложное занятие. Если подряд идет несколько лекционных занятий, то усталость только увеличивается. Как следствие, уменьшается концентрация во время работы с ведомостью, и возрастает вероятность ошибиться.

Когда преподаватель выполняет проверку посещаемости всей группы, ошибиться довольно сложно, потому что столбец с посещаемостью занятия заполняется последовательно сверху вниз.

Иначе обстоит ситуация, когда преподаватель отмечает опоздавшего студента (особенно, если он уже не первый опоздавший). Преподаватель может ошибиться, сопоставляя ячейку посещаемости с фамилией студента. Вероятность ошибки возрастает, если на данный момент отсутствует много студентов, которые по списку находятся рядом с опоздавшим (выше и ниже). Вероятность ошибки также прямо пропорционально зависит от расстояния между ячейкой студента и ячейкой посещаемости.

В интерфейсе электронной ведомости при добавлении нового занятия можно сделать возможность указать вариант по умолчанию для посещаемости студентов группы (например, все присутствуют). Таким образом, при проверке посещаемости для многих студентов будет уже выбран подходящий вариант по умолчанию (например, студент присутствует на занятии).

Поскольку все варианты посещаемости, как правило, отмечаются одним цветом, сложно оценить статистику посещаемости, взглянув на ведомость.

## Анализ интерфейсов программ-аналогов

Для ведомостей посещаемостей в приложении будет табличное представление. Рассмотрим приложения для ОС Android, использующие табличное представление для отображения данных. На данный момент устоявшихся русскоязычных названий для терминов мобильных устройств нет, поэтому ниже приведен список переводов используемых терминов.

* Spinner – элемент управления в OC Android, похожий на выпадающий список.
* Action Bar – панель в верхней часть экрана приложения в OC Android. Может содержать различные элементы управления.
* Time Picker – элемент управления в ОС Android для выбора времени.
* Touch – коснуться экрана и отпустить (жест).
* Long touch – коснуться экрана, подождать, отпустить (жест).
* Swipe (Drag) – коснуться экрана, подвинуть палец, отпустить (жест).

### Google Календарь

Google Календарь – многофункциональный органайзер. На рис. 2.1 можно видеть табличное представление запланированных дел. Из элементов управления можно отметить Spinner в Action Bar, переключающий формат представления таблицы (неделя, месяц, повестка дня), кнопку «TODAY», по нажатию на которую табличное представление переходит на текущую дату.

Табличное представление имеет закрепленную сверху строку с заголовками таблицы и закрепленный слева столбец с названиями строк. Табличное представление прокручивается вертикально и горизонтально с помощью жеста Swipe. Вызвать диалог добавления нового запланированного дела можно с помощью жеста Touch на пустой ячейке.

Из интересных интерфейсных решений можно отметить использование Time Picker для выбора времени при добавлении нового дела (рис 2.2). Данный элемент управления упрощает выбор времени и уменьшает возможность ошибиться при выборе (по сравнению с обычным текстовым полем).

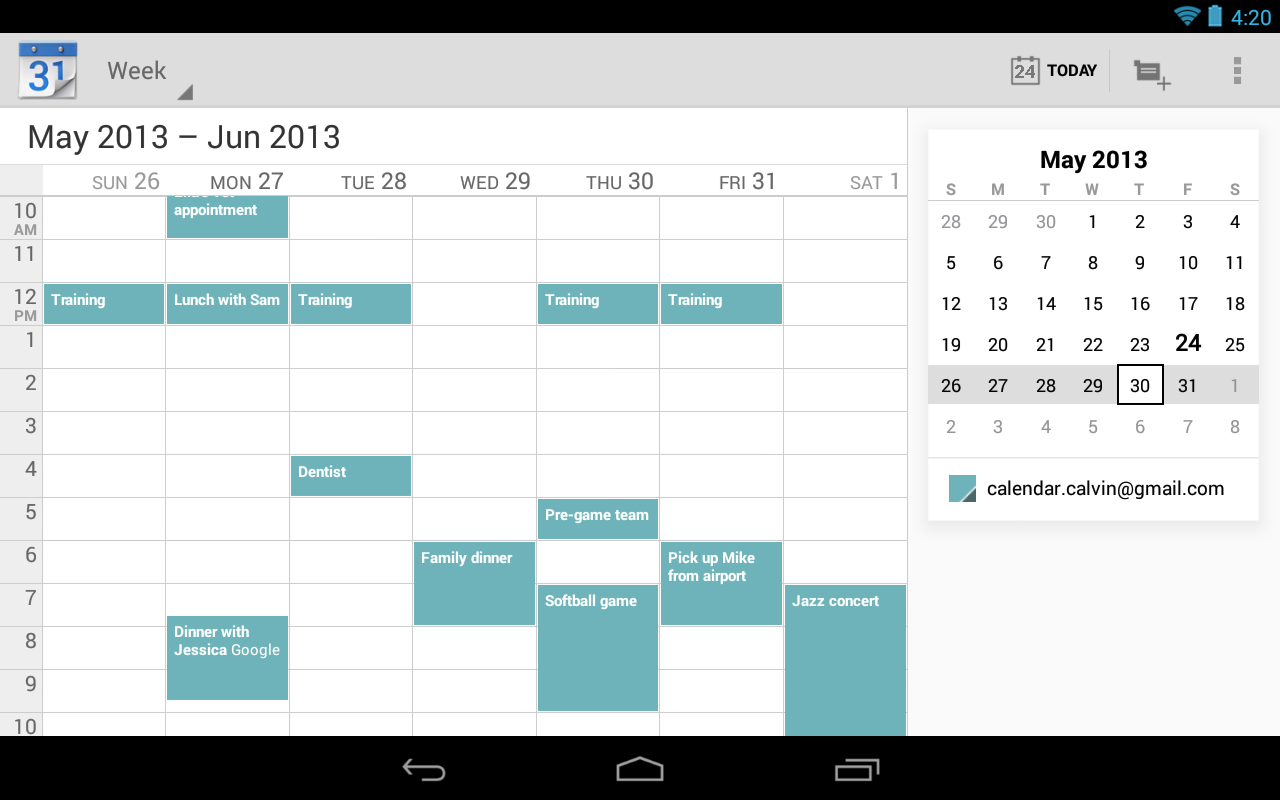


Рис. . Скриншот приложения Google Календарь

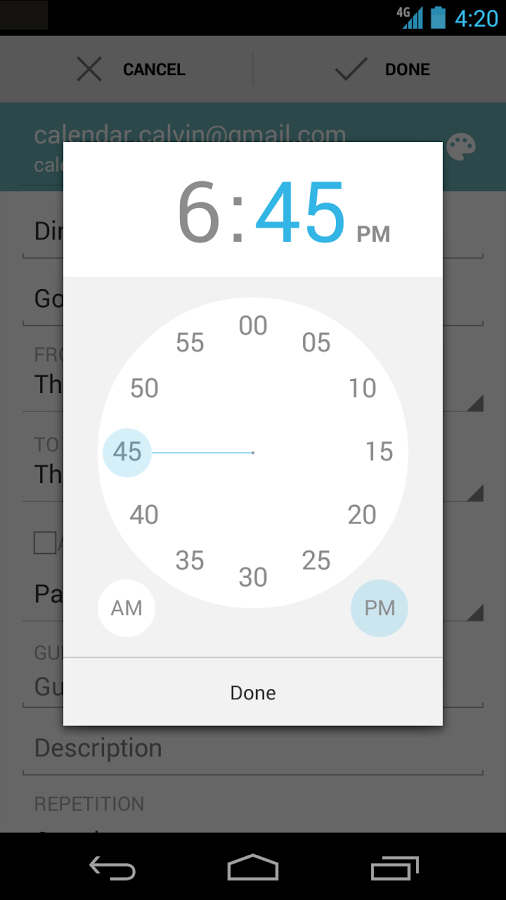


Рис. . Диалог выбора времени в Google Календарь

### Google Таблицы

Google Таблицы – веб-приложение, реализующее функции табличного процессора. Данное веб-приложение имеет также мобильный интерфейс, на который стоит обратить внимание.

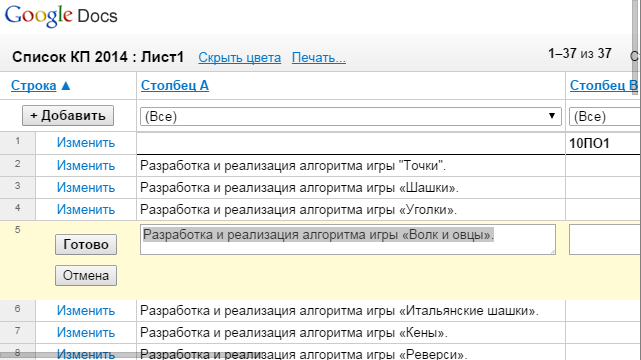


Рис. .. Редактирование строки документа в мобильной версии Google Таблицы

Табличное представление в данном приложении имеет поведение, подобное «Google Календарь». Главным отличием является редактирование строки. Нажатие на кнопку «Изменить» в данном случае приводит к началу редактирования строки таблицы. В этом состоянии интерфейса можно изменять содержимое ячеек строки. Для сохранения изменений или отмены предусмотрены соответствующие кнопки.

Еще одним отличием являются индикаторы прокрутки таблицы справа и снизу.

### Оценка актуальности использования интерфейсных решений программ-аналогов в проектируемом интерфейсе

Рассмотрим, какие из особенностей интерфейсов программ-аналогов подходят для использования в проектируемом интерфейсе. Оценка актуальности представлена в табл. 2.1.

Таблица 2.1

**Оценка актуальности особенностей интерфейсов программ-аналогов**

|  |  |
| --- | --- |
| **Особенность интерфейса** | **Актуальность** |
| Зафиксированные первые строка и столбец в таблице | Нужно использовать |
| Использование жеста Swipe для прокручивания табличного представления | Нужно использовать |
| Переключение формата представления таблицы (Google Календарь) | Не актуально |
| Кнопка перехода на актуальное представление в Action Bar | Не актуально |
| Вызов диалога редактирования ячейки на жест Touch (Google Календарь) | Диалог не подходит для редактирования посещаемости – слишком много действий. |
| Вызов режима редактирования строки на нажатие кнопки «Изменить» (Google Таблицы) | Также не подходит для редактирования посещаемости – много действий |
| Использование индикаторов для отображения текущей прокрутки таблицы (Google Таблицы) | Индикаторы не нужны, ориентироваться можно по строке и столбцу (как в Google Календарь) |
| Использование Time Picker (Google Календарь) | Данный элемент не нужен, но можно сделать что-то подобное для редактирования посещаемости |

Таким образом, часть интерфейсных решений в программах-аналогах можно использовать.

## Формирование контрольного списка требований к проектируемому интерфейсу

Рассмотрев особенности работы с ведомостями посещаемости и существующие интерфейсные решения для работы с табличным представлением данных можно составить контрольный список требований к проектируемому интерфейсу.

1. Приложение должно соответствовать Android Guidelines [?] (0,3).
2. Табличное представление ведомости (0,3)
   1. Данные ведомости должны быть представлены в табличной форме (0,03).
   2. Первый столбец должен содержать порядковые номера и ФИО студентов и должен быть закреплен слева (0,03).
   3. Первая строка должна содержать даты занятий и должна быть закреплена вверху (0,03).
   4. Последняя строка должна содержать заметки и быть закреплена снизу (0,03).
   5. Ячейки с посещаемостью должны быть выделены разными цветами в зависимости от типа посещения. Цвета должны приятно выглядеть и контрастировать между собой (0,03).
   6. Должны отображаться название предмета и группа, для которой выбрана ведомость (0,03).
   7. Должна быть возможность добавлять и редактировать заметки для занятий (0,03).
   8. Должна быть возможность переключать ведомости групп для текущего занятия (0,03).
   9. Добавление нового занятия (0,03)
      * Должна быть возможность добавлять новые занятия в ведомость (0,01).
      * Должна быть возможность выбрать дату нового занятия при добавлении (0,01).
      * Дата нового занятия по умолчанию должна быть выставлена на сегодняшнее число (0,005).
      * Должна быть возможность выбора варианта по умолчанию для посещаемости студентов (0,005).
   10. Редактирование занятия должно начинаться с помощью жеста Touch на ячейку с датой занятия (0,03).
3. Редактирование посещаемости (0,4)
   1. Нужен вариант редактирования, в котором возможность ошибиться при выборе варианта посещаемости для студента минимальна (0,2).
   2. При редактировании должен отображаться вариант, выбранный по умолчанию (0,05).
   3. При выборе варианта пользователем, этот вариант должен отображаться тоже (0,05).
   4. Должен быть удобный способ навигации по списку студентов (0,1).

Формирование контрольного списка завершено, и можно начинать проектирование интерфейса ведомостей лекционных занятий для мобильного приложения для ОС Android.

# КОНСТРУКТОРСКАЯ ЧАСТЬ