**Quick Homework**

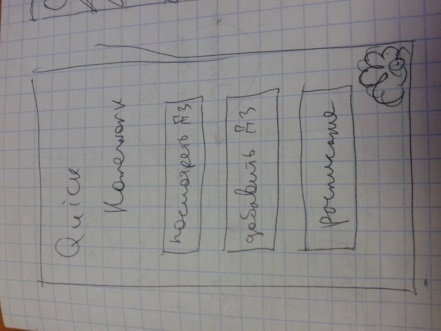
**Design – document**

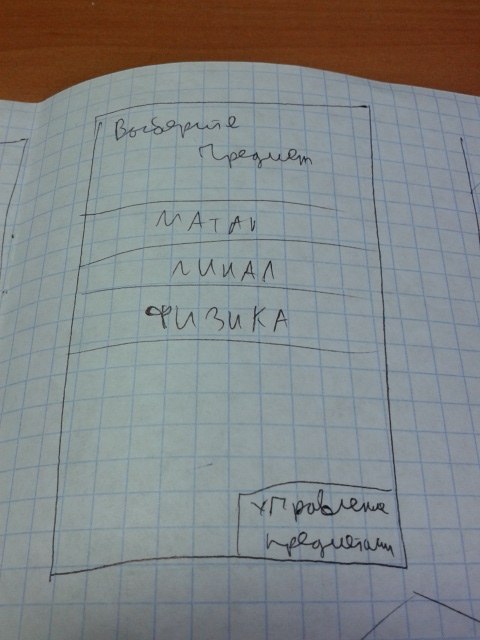
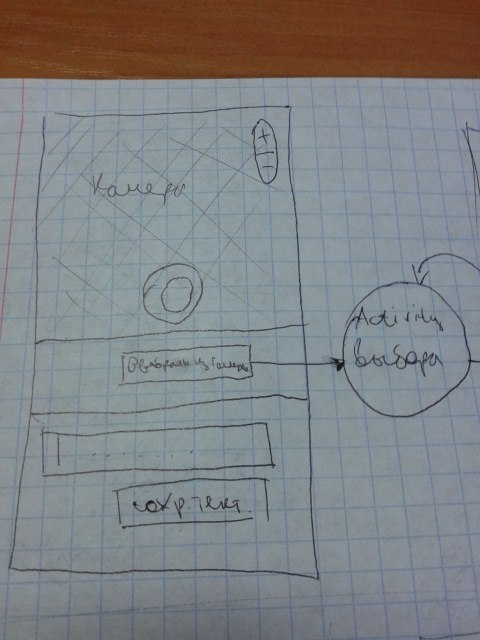
**1. Общее описание.**

Приложение на андроид для студентов и школьников позволяет максимально быстро сохранять и просматривать ДЗ. Сделав два нажатия, вы увидите ваше ДЗ по выбранному предмету. За 3 нажатия вы сможете сфотографировать ДЗ по выбранному предмету, выбрать фото из библиотеки и дополнительно (или отдельно) сделать текстовую заметку. Хранится 3 последних ДЗ. Плюс есть возможность просматривать расписание за 1 нажатие внутри программы. И менять его за 3 нажатия.

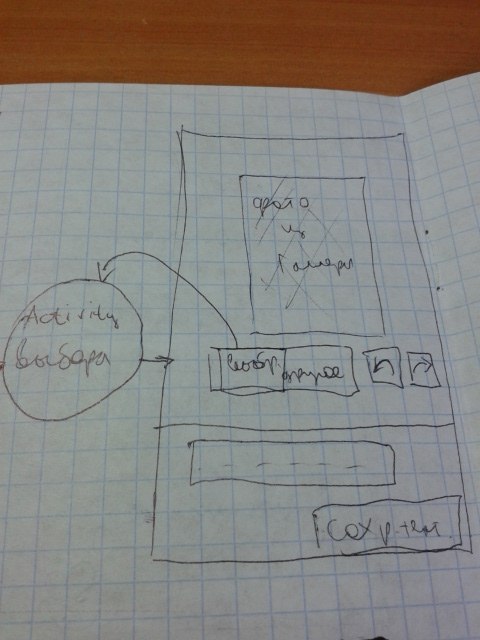
**2. Подробное описание.**

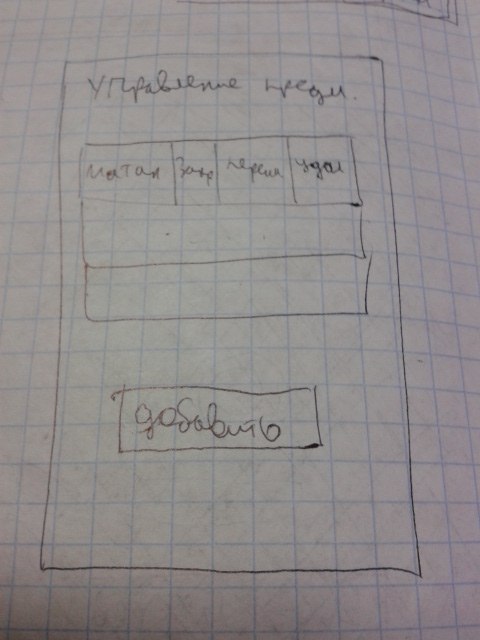
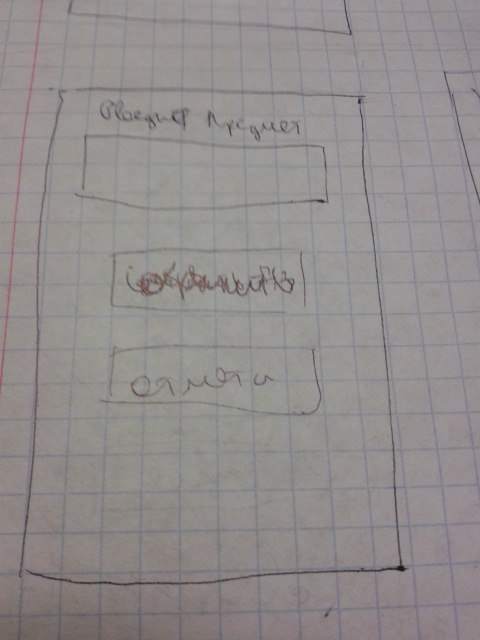
Здесь будет описание каждой активность приложения.

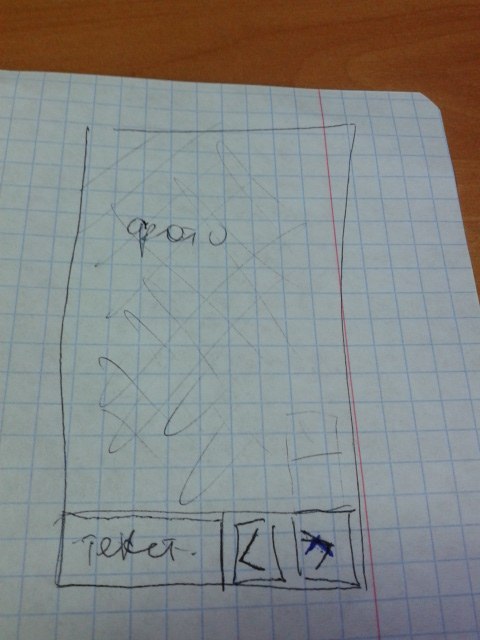
**1. MainActivity** – главное меню приложения, содержит название и 3 кнопки, переводящие на соответствующие активности.  
- Посмотреть ДЗ  
- Добавить ДЗ  
- Расписание

**2. ChooseLessonActivity** – экран выбора предмета, запускающийся при нажатии на кнопки «Посмотреть» и «Добавить». Содержит список предметов и, если экран предшествует добавлению, то в углу кнопка «Управление предметами». Наша цель – выбрать предмет, после чего происходит переход на нужный экран.

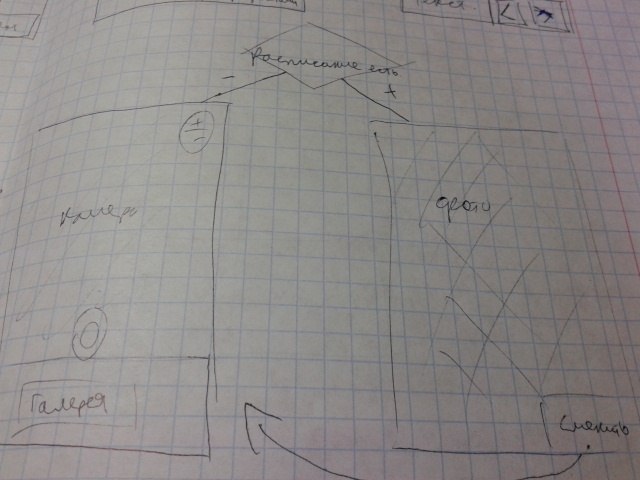
**3. AddActivity –** активити, которое выходит после нажатия «Добавить ДЗ» и выбора предмета. Состоит из окна с камерой (на котром есть кнопки фотографирования и зума), кнопки «Выбрать из галереи», и внизу текстового поля ввода и кнопки «Сохранить». При делании снимка, он сохраняется, после этого можно выйти ( на хард кнопки), либо добавить заметку в поле и сохранить текст, можно просто сохранить текст и не делать фото. Альтернативой фотографированию является кнопка «Выбрать из галлереи», по нажатии на которую андроид переносит нас в галлерею и мы выбираем файл, далее запускается следующая активность.

**4. FromGalleryActivity –** выходит после выбора картинки из галлереи для сохранения ДЗ. Выбранное фото выводится на экран, мы можем выбрать другое фото, повернуть это, все это кнопками внизу. Под этим находится панель для ввода текста аналогично предыдущему окну, т.к. выбор из библиотеки просто альтернатива фотографированию.

**5. ManageSubjectsActivity –** панель управления предметами. Запуск ведется из страницы списка предметов при добавлении предмета. На экране список предметов, для каждого имеются кнопки «Изменить», «Удалить» и CheckBox «Закрепить». Внизу большая кнопка «Добавить». Итак, галочка «Закрепить» нужна для закрепления предмета наверху, сначала выводятся закрепленные предметы, затем незакрепленные (а так по алфавиту). Кнопка «Удалить» выводит диалог для подтверждения удаления, при удалении происходит удаление предмета и связанных с ним фото на карте телефона. Кнопки «Добавить» и «Изменить» переводят на следующую активность.

**6. AddChangeSubjectActivity** – окно для добавления нового предмета, либо переименования существующего. Тектовое поле для ввода названия и кнопки «Сохранить» и «Отмена». Если мы зашли для смены предмета, то в текстовом поле сразу введено название предмета. Нажатие сохранить сохраняет данные и переводит обратно на окно управоения предметами.

**7. ShowActivity –** окно просмотра ДЗ по выбранному предмету. На экране фото, которое можно зумить пальцами, внизу текст, если он сохранен и кнопки для переключения между предыдущими ДЗ. Если нет текстового поля или фото, то они не отображаются.

**8. ShowScheduleActivity –** окно просмотра расписания, если оно уже есть. Если оно еще не добавлено, то запускается следующая активность, как показано на схеме. На окне просто фото, которое можно зумить и поверх него кнопка «Сменить», запускающая след. активность.

**9. ChangeScheduleActivity** – активити для добавления расписания. Похоже на AddActivity, за исключением наличия поля для ввода текста, в остальном все идентично.

**3. Проектирование архитектуры.**

Здесь подробно опишу, как буду реализовывать проект, все классы, основные методы и т.д. Описание верстки дизайна не приведено т.к. оно тривиально и вытекает из описания в предыдущей главе.

**Часть 1. Классы – помощники.**

**1. Subject** – класс предмета, 3 публичных поля – id, name, isLocked, int countOfHomeWorkIs, String[] textHomeWork.

**2. Repository** – класс инкапсулирует логику взаимодействия с БД. Он должен предоставлять следующие методы  
 ArrayList<Subject> GetSubjects() – возвращает список предметов, упорядоченных сначала по признаку закрепленности, затем по алфавиту  
 Subject GetSubject(int id)  
 SaveTextHomeWork(int id);  
 AddSubject(String Name)  
 SetLock(int id, boolean isLocked)  
 DeleteSubject(int id)  
 RenameSubject(int id, String Name)  
 GetPathToHomeTask(int id, int number) – возвращает путь у файлу с ДЗ (number – это номер ДЗ. Последнее – это 0).  
 Реализацию опускаю, для помощи используем приложение SpportStore.

**3. ManageSubjectsAdapter** – адаптер списка на панели управления предметами.  
 В конструкторе получаем список предметов. Заполняем список, в тэг кнопок суем позицию. Пишем обработчик каждой кнопки, как их сделать очевидно.

**Часть 2. Активити.**

**1. MainActivity.** Элементарная активность. Просто находим кнопки и запускаем другие активности.

**2. ChooseLessonActivity** – во первых интент должен передать для чего вызывается активность (просмотр или добавление). Затем, в зависимости от этого меняется экран в плане удаления кнопки «Управление предметами». Затем получаем от репозитория спиок товаров. Создаем обыкновенный список. Припичываем обработчик выбора, который будет запускать нужное активити, передавая ему через интент id предмета.

**3. AddActivity.** Здесь работа с камерой, что сильно усложняет активность. Но я не унываю. Итак, лаяут сверстан. Сначала обработаем все, что не связано с камерой. Кнопка «Галлерея», запускает активити на результат, в результате берем путь, суем в интент и открываем следующее активити. Кнопка сохранить сует в репозитория текст из поля ввода (если там пусто, выводим инфо-сообщение об этом). На этом все, что не связано с камерой. Все, что связано с камерой копируем из тестового приложения. Обрабатываем кнопки фото и зума, при деланьи фото, выводим инфо-сообщение и сообщаем об этом репозиторию.

**4. FromGalleryActivity.** Сначала обнаружим все вью и дадим обработчики в т.ч. для кнопки «сохранить» копируем код из предыдущего активити, делать отдельный класс не имеет смысла. Кнопки поворота не трогаем.Из интента достаем путь к выбранной фотке. Создаем битмап, сохраняем фото. Теперь обработаем кнопки поворота. Поворачиваем битмап, сразу перезаписываем файл.

**5. ManageSubjectsActivity** – здесь вступает в игру наш адаптер. Мы заполняим им спиок, сами обрабоываем кнопку «Добавить» и все.

**6. AddChangeSubjectActivity** – все просто, проверяем нет ли чего в интенте, если есть, суем имя в поле. При отмене возвращаемся, при сохранении суем в репозиторий, выводим инфо-сообщение и возващаемся.

**7. ShowActivity** – достаем из интента номер предмета, получаем путь к фото, загружаем, выводим изображение. Загружаем из БД текст. Если он есть заполняем текствью. Находим кнопки стрелок. Обрабатываем их. В них меняем номер домашки и вызываем тот же метод, что при старте.

**8. ShowScheduleActivity** – как видно из схемы, сначала проверяем есть ли у нас расписание, если нет, то запускаем следующее активити. В ообщем, если остались здесь, то загружаем картинку, выводим ее, обрабатываем кнопку «Сменить».

**9. ChangeScheduleActivity and FromGalleryScheduleActivity** – здесь полностью копируем код и активити 3 и 4.

**4. План разработки.**

1. Главное меню  
 -верстаем лаяут.  
 -делаем активити.

2. Побочные классы.  
 -пишем класс предмета.  
 -пишем репозиторий, тестим

3. Добавление.  
 -Пишем активити списка.  
 -Верстаем лаяут.  
 -Делаем все, кроме камеры и галереи  
 -Добавляем камеру, тестим.  
 -верстаем лаяут  
 -Пишем активити после галереи.  
Прим. пока тестирование просто через просмотр картинок через диспетчер файлов

4. Менеджмент.  
 -верстаем лаяуты  
 -пишем адаптер  
 -верстаем лаяут добавления  
 -пишем это активити

5. Просмотр ДЗ.  
 -Верстаем лаяут  
 -Вставляем фото  
 -Выбираем из все ДЗ

6. Расписание.  
 -Лаяут фотонья и галереи  
 -Активити изменения расписания  
 -Активити просмотра расписания

7. Бета-тест.

8.Прилизывание.  
 -Инфо от разработчиков  
 -Рефакторинг кода  
 -Статистика

9. Публикация.

**5. Итоги.**

Я закончил. Писал 2 дня без перерывов. Вот статистика

Строк – 2000.

Часов – 23.

Строк в час – 87.

Производительность (кода в день с учетом проектирования) – 500.

Вообщем, устал. Времени не было вообще. Писал, как бешеный. Цель была – написать. О коде не думал. Но судя по статистике общая производительность лишь чуть выше, чем у спорт магазина. Но это отчасти потому, что много времени было уделено дизайну. Все пора публиковать.