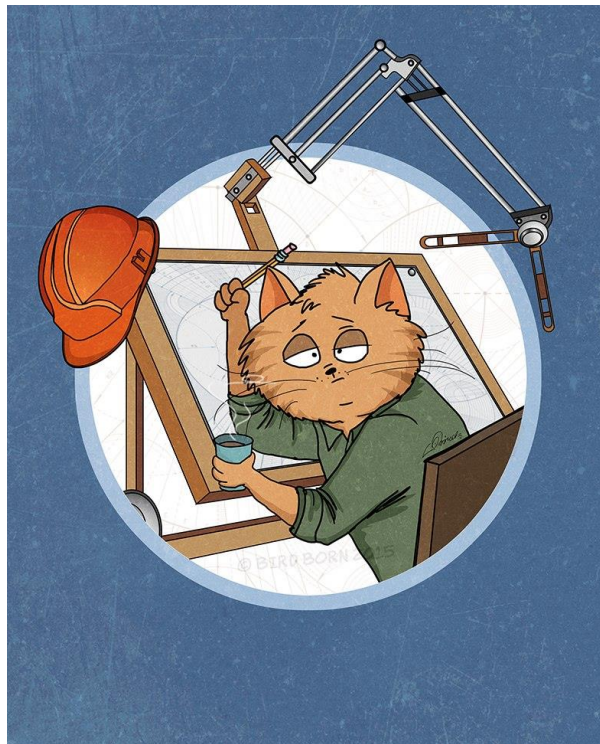


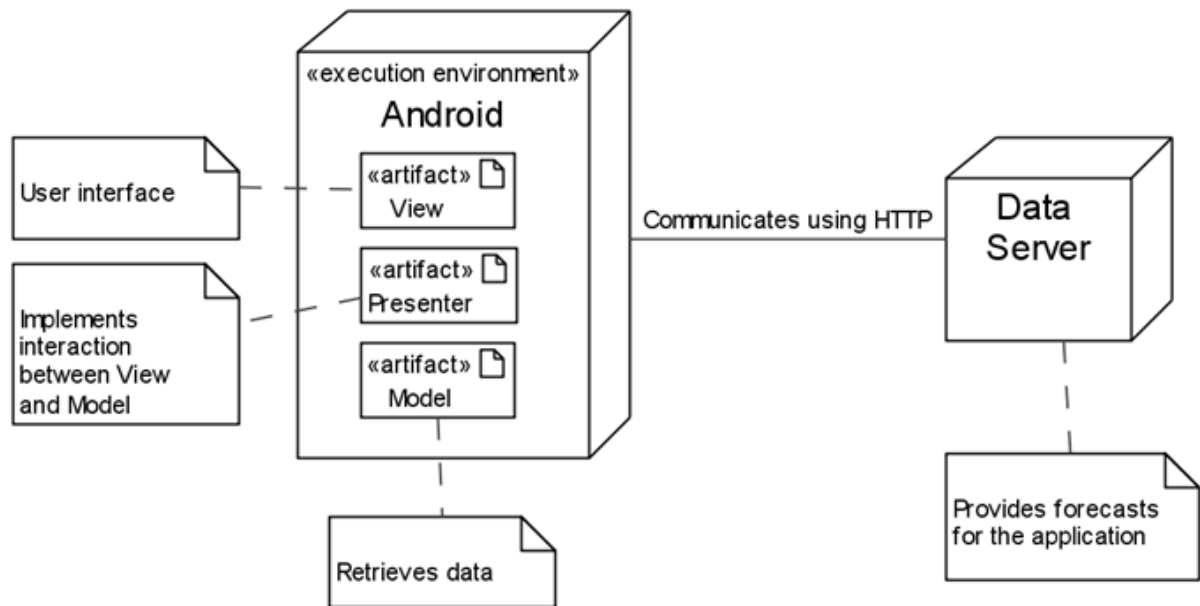
Лабораторная работа №4
Исследование архитектурного решения
Котики



TO BE

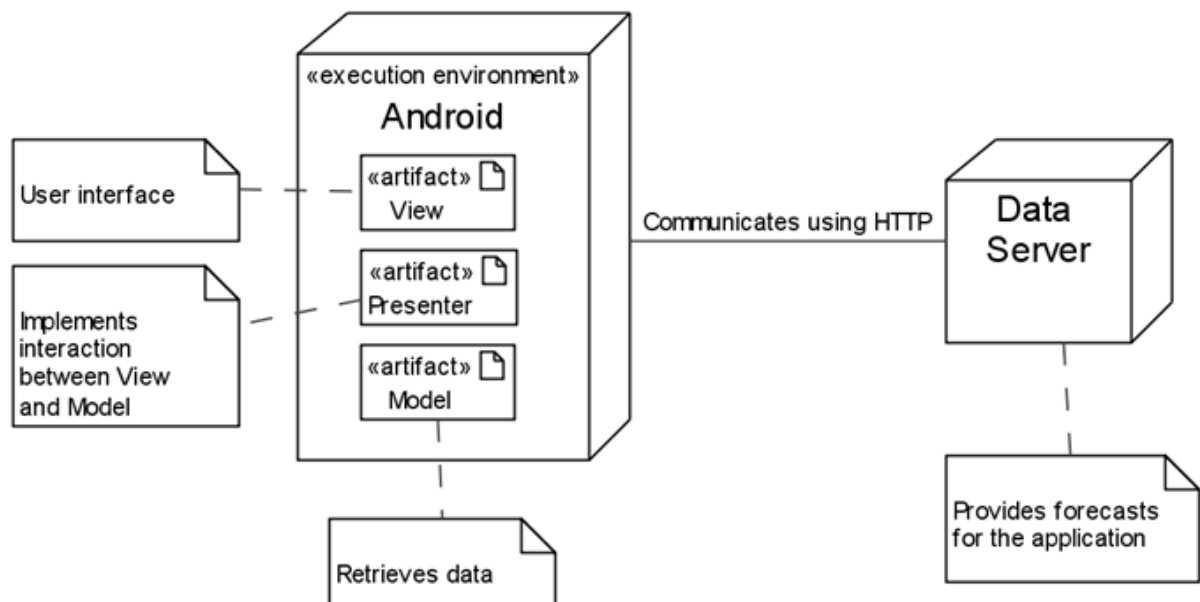
1. Приложения для мобильных устройств. (Android)
2. Двухуровневая стратегия развертывания.
3. Приложение написано на Java, так как данный язык имеет огромную популярность и имеет полноценную документацию и большое количество примеров. При передаче данных с сервера выбрали формат JSON: очень просто из него извлекать данные.
4. Основным показателем качества:
 - a. Удобство и простота использования: Малое количество функций (нельзя “загромождать” приложение), использование общепринятых способов взаимодействия с приложением (управление свайпами). Критерии приемки: доступ к любой части приложения не больше чем за 3 свайпа.
 - b. Надёжность. В приложении не будет критических сбоев при нестабильности / отсутствии интернет соединения.
 - c. Доступность. Приложение доступна на любых мобильных телефонах с версией Android 7.0 и выше.
5. Пути реализация сквозной функциональности: управление конфигурациями. При изменении настроек приложения, изменения произойдут одновременно на всех уровнях.

6. Обобщенное представление архитектуры

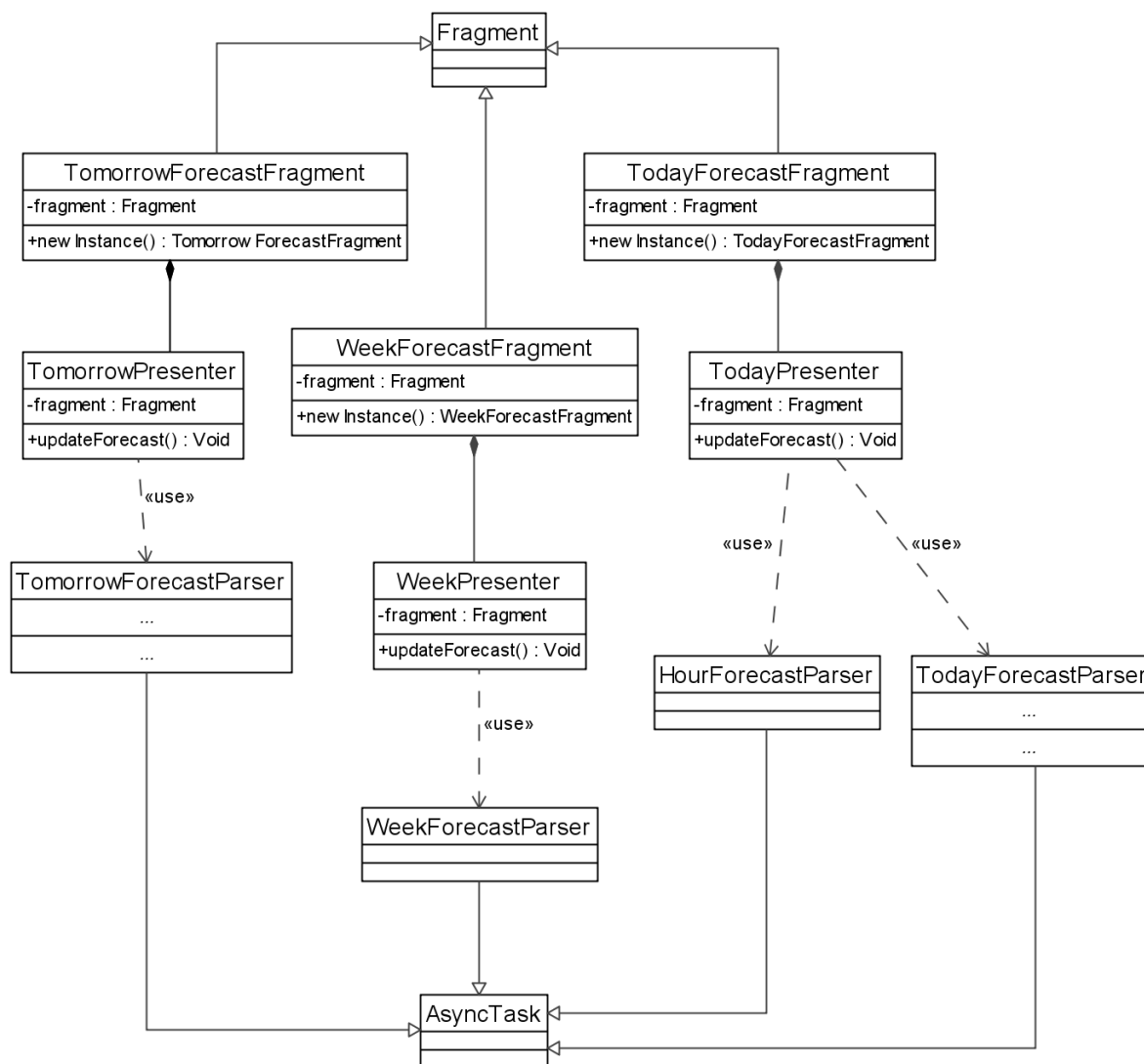


AS IS

1. В архитектуре приложения четко просматривается паттерн MVP
2. Обобщенное представление архитектуры



3. Диаграмма классов



TO BE

В целом архитектура приложения соответствует заявленной. Однако, ходе разработки были допущены некоторые ошибки, которые стали причиной нарушения MVP-архитектуры.

Ошибки:

Model напрямую выводит данные во View, чего в MVP-архитектуре быть не должно. Во многом из-за этого Presenter получился практически пустым.

Как исправлять:

Необходимо произвести рефакторинг слоев Model и Presenter. Нужно сделать так, чтобы слой Model передавал данные слою Presenter, а Presenter в свою очередь выводил их во View.