

Разработка мобильного клиента HelpDesk системы

Н.Ю Береснев

Проект ГПО АОИ-1601 «Модуль автоматизации работы операторов поддержки для Help-desk систем»

*Научный руководитель:
преподаватель кафедры АОИ
Голубева А. А.*

Введение

Первоначально мобильные приложения использовались для быстрой проверки электронной почты, но их высокий спрос привел к расширению их назначений и в других областях. [1] Использование мобильного клиента для Helpdesk системы позволяет вывести её на новый уровень, поскольку оператор может в любой момент получить уведомление о поступившей заявке и приступить к работе прямо со своего смартфона. Однако, написание качественного программного продукта - не простая задача: программный продукт должен не только удовлетворять функциональным требованиям, но и быть удобным в сопровождении, а также «гибким» к дальнейшим изменениям функционала.

Шаблон проектирования MVP

В настоящий момент существует множество шаблонов проектирования Android приложений, один из которых - MVP (Model View Presenter). Данный шаблон проектирования позволяет создавать абстракцию представления. Для этого необходимо выделить интерфейс представления с определенным набором свойств и методов. Презентер, в свою очередь, получает ссылку на реализацию интерфейса, подписывается на события представления и по запросу изменяет модель. [2] Схематически это можно отобразить следующим образом: (рисунок 1)

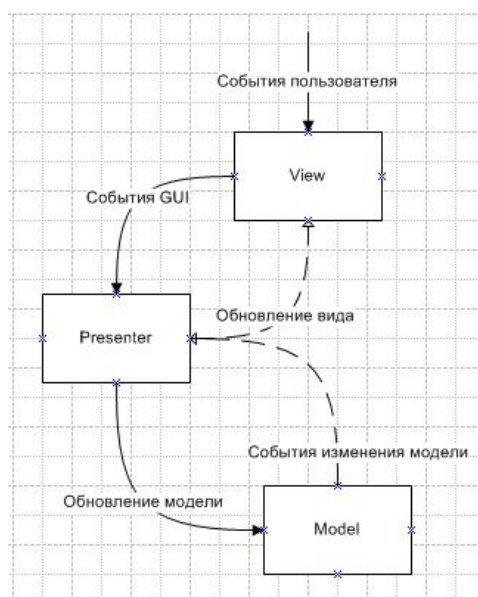


Рисунок 1 - Шаблон проектирования MVP

Проектирование мобильного клиента с использованием шаблона MVP

Выделив все функциональные требования, следующим этапом было проектирование системы. Поскольку в последнее время шаблон проектирования MVP пользуется особой популярностью, было принято решение проектировать систему с его использованием. В результате работы программный продукт имел следующую архитектуру (рисунок 2).

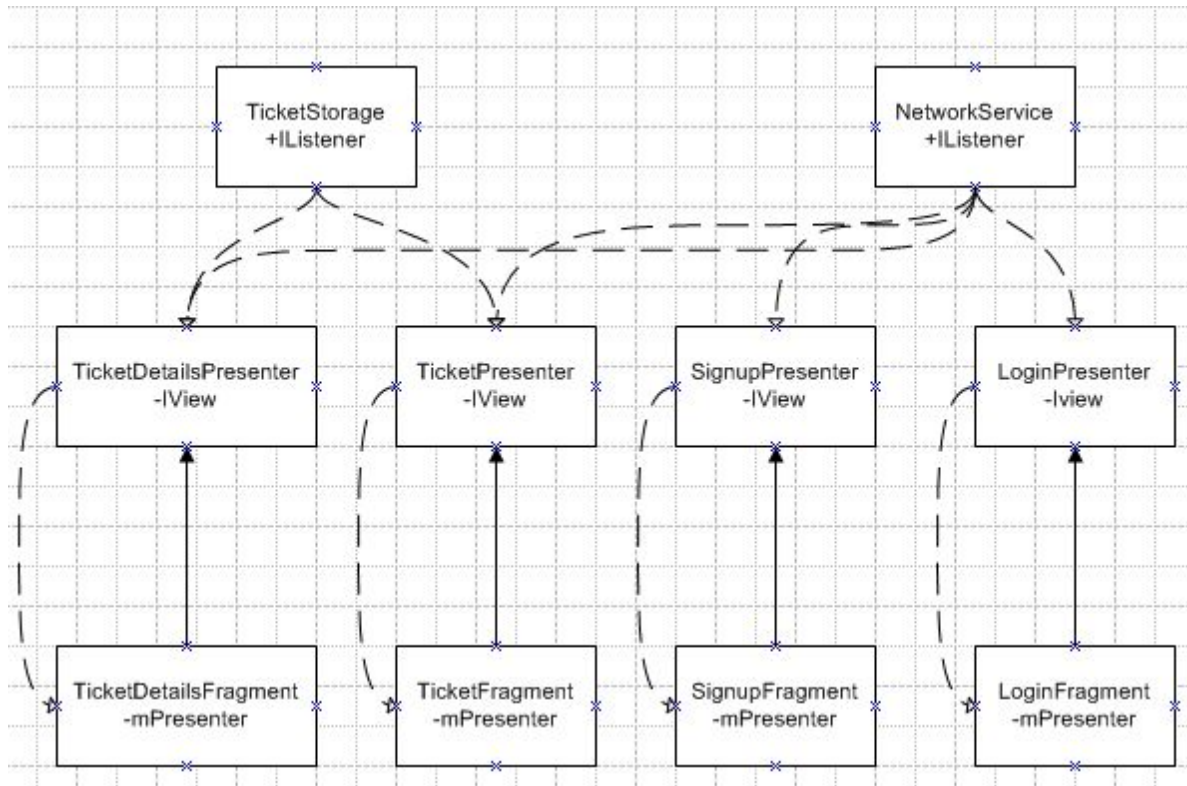


Рисунок 2 - Архитектура мобильного клиента Helpdesk-системы

Клиент состоит из четырех фрагментов, которые являются отображениями:

1. LoginFragment - фрагмент, предназначенный для авторизации оператора в системе.
2. SignupFragment - фрагмент, предназначенный для регистрации нового оператора.
3. TicketFragment - фрагмент, на котором отображаются все поступившие заявки оператору.
4. TicketDetailsFragment - фрагмент для работы с конкретной заявкой.

В качестве модели выступают TicketStorage - это хранилище всех поступивших заявок, и NetworkService - сервис, который реализует все сетевые взаимодействия.

Каждому из фрагментов соответствует свой презентер, который реализует логику по обновлению модели и отображений. Следовательно, в результате такого подхода была достигнута гибкая архитектура, поскольку клиент не зависит от отображений в любой момент времени - вся логика реализована на стороне презентеров. Это позволяет спокойно изменять отображения и наращивать функционал приложения без особых проблем.

Выводы

Правильно спроектированная архитектура программного продукта позволяет избежать множества проблем как на этапе разработки, так и во время его сопровождения. В результате работы был спроектирован и реализован мобильный клиент Helpdesk-системы с использованием шаблона MVP.

Использованная литература

1. Мобильное приложение [Электронный ресурс] - Режим доступа https://ru.wikipedia.org/wiki/Мобильное_приложение.
2. Паттерны для новичков [Электронный ресурс] - Режим доступа <https://habrahabr.ru/post/215605>.