ПРИЛОЖЕНИЕ Е. ТЕКСТ ПРОГРАММЫ

АННОТАЦИЯ

В данном программном документе приведен текст программы на разработку мобильного приложения для курьеров службы курьерской доставки «DP Stuff Provider».

В данном программном документе, в разделе «Текст программы» указан текст программы и его описание.

СОДЕРЖАНИЕ

1. ОБЪЕКТ ИСПЫТАНИЙ

1.1. Наименование программы

Наименование - «DP Stuff Provider».

1.2. Область применения программы

Программа предназначена к применению в профильных подразделениях ООО «DP Stuff Provider».

1.3. Модули

Таблица 1 - Модули программы

№	Модуль	Описание	Кол-во строк	Размер в КБ
1	MainActivity.kt	Окно авторизации	77	2.56
2	MainMenu.kt	Главное меню	51	1.74
3	MoreInfo.kt	Окно просмотра подробной информации	87	3.63
4	AccountFragment.kt	Фрагмент личного кабинета	47	1.81
5	ActiveDeliveryFragment.kt	Фрагмент активных заказов	47	1.72
6	AllDeliveryFragment.kt	Фрагмент доступных заказов	55	1.87
7	DialogFinishOrderFragment.kt	Фрагмент всплывающего окна завершения заказа	66	2.76
8	OrderComposAdapter.kt	Адаптер для заполнения списка товаров, входящих в заказ	40	1.54
9	OrdersAdapter.kt	Адаптер для заполнения списка заказов (доступных и активных)	80	3.10

1.4. Код программы

1) MainActivity.kt

```
package com.example.dpstuffprovider

import android.content.Context
import android.content.Intent
import android.content.SharedPreferences
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.view.View
import android.widget.Toast
import com.example.dpstuffprovider.api.ApiService
import com.example.dpstuffprovider.interfaces.ICouriers
import com.example.dpstuffprovider.models.CouriersData
import kotlinx.android.synthetic.main.activity_main.*
import retrofit2.*
import retrofit2.converter.gson.GsonConverterFactory

/**
```

```
 * Стартовое активити с формой авторизации
 */
class MainActivity : AppCompatActivity() {

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        val sp : SharedPreferences = getSharedPreferences("sp", Context.MODE_PRIVATE)
        if(sp.getBoolean("isLogin", false)){
            login(sp.getString("login", "")!!, sp.getString("password", "")!!)
        }
    }

    override fun onRestart() {
        super.onRestart()
        progressBarOff()
        getSharedPreferences("sp", Context.MODE_PRIVATE).edit().putBoolean("isLogin",
false).apply()
    }

    fun login(login : String, password : String){
        progressBarOn()

        ApiService().getCouriers(login, password) {
            if(it != null){
                val courier = it[0]

                Toast.makeText(applicationContext, "Здравствуйте, ${courier.firstName}",
Toast.LENGTH_LONG).show()

                val editor : SharedPreferences.Editor = getSharedPreferences("sp",
Context.MODE_PRIVATE).edit()
                editor.putBoolean("isLogin", true)
                editor.putString("login", login)
                editor.putString("password", password)
                editor.apply()

                val intent : Intent = Intent(applicationContext, MainMenu::class.java)
                intent.putExtra("courier", courier)
                startActivity(intent)
            }
            else{
                progressBarOff()
                Toast.makeText(applicationContext, "Неверный логин и/или пароль",
Toast.LENGTH_LONG).show()
            }
        }
    }

    fun onClickLogin(view : View){
        login(etLogin.text.toString(), etPassword.text.toString())
    }
    fun progressBarOn(){
        progressBar.visibility = View.VISIBLE
        btnLogin.visibility = View.GONE
    }

    fun progressBarOff(){
        progressBar.visibility = View.GONE
        btnLogin.visibility = View.VISIBLE
    }
}
```

2) MainMenu.kt

```
package com.example.dpstuffprovider

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import androidx.fragment.app.Fragment
import com.example.dpstuffprovider.fragments.AccountFragment
import com.example.dpstuffprovider.fragments.ActiveDeliveryFragment
import com.example.dpstuffprovider.fragments.AllDeliveryFragment
import com.example.dpstuffprovider.models.CouriersData
import kotlinx.android.synthetic.main.activity_main_menu.*

/**
 * Основное активити с навигационным меню
 */
class MainMenu : AppCompatActivity() {

    var courier : CouriersData? = null

    /*val allDeliveryFragment = AllDeliveryFragment()
    val activeDeliveryFragment = ActiveDeliveryFragment()
    val accountFragment = AccountFragment()*/

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main_menu)

        bottomNavMenu.setOnNavigationItemSelectedListener {
            when(it.itemId){
                R.id.allDelivery ->
makeCurrentFragment(AllDeliveryFragment())
                R.id.activeDelivery ->
makeCurrentFragment(ActiveDeliveryFragment())
                R.id.account -> makeCurrentFragment(AccountFragment())
            }
```

```
                  true
              }

              courier = intent.getSerializableExtra("courier") as CouriersData?
              //nameCourier.text = "${courier.lastName} ${courier.firstName}
${courier.patronymic}"
          }

          fun makeCurrentFragment(fragment: Fragment) =
              supportFragmentManager.beginTransaction().apply {
                  replace(R.id.frameWrapper, fragment)
                  commit()
              }

          override fun onResume() {
              super.onResume()

              makeCurrentFragment(AllDeliveryFragment())
          }
      }
```

3) MoreInfo.kt

```
package com.example.dpstuffprovider

import android.app.Dialog
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.util.Log
import android.view.View
import android.widget.Toast
import androidx.fragment.app.FragmentManager
import androidx.recyclerview.widget.LinearLayoutManager
import com.example.dpstuffprovider.adapters.OrderComposAdapter
import com.example.dpstuffprovider.api.ApiService
import com.example.dpstuffprovider.fragments.DialogFinishOrderFragment
import com.example.dpstuffprovider.models.ClientData
import com.example.dpstuffprovider.models.CouriersData
import com.example.dpstuffprovider.models.OrderCompos
import com.example.dpstuffprovider.models.OrdersData
import kotlinx.android.synthetic.main.activity_more_info.*
import java.lang.Exception

/**
 * Активити с подробной информацией о выбранном заказе
 */
class MoreInfo : AppCompatActivity() {

    var orderInfo: OrdersData? = null
    var courier: CouriersData? = null
    var clientInfo: ClientData? = null

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_more_info)
        try{
            orderInfo = intent.getSerializableExtra("orderInfo") as OrdersData?
            courier = intent.getSerializableExtra("courier") as CouriersData?
            tvAddress.text = orderInfo!!.address
            tvFrontDoor.text = "Подъезд: ${orderInfo!!.frontDoor.toString()}"
            tvFloorNum.text = "Этаж: ${orderInfo!!.floorNum.toString()}"
            tvAppartmentNum.text = "Квартира: 
${orderInfo!!.apartNum.toString()}"
            tvIntercom.text = "Домофон: ${orderInfo!!.intercom}"
            clientInfo = intent.getSerializableExtra("clientInfo") as 
ClientData?
            tvClientName.text = "${clientInfo!!.firstName} 
${clientInfo!!.patronymic}"
            tvClientPhone.text = "Телефон: ${clientInfo!!.phone}"

            btnCheckMap.setOnClickListener {
                Toast.makeText(applicationContext, "Скоро станет доступным...", 
Toast.LENGTH_LONG).show()
            }

            if(orderInfo!!.status == "Выдан курьеру"){
                btnEndOrder.visibility = View.VISIBLE
                btnConfirmOrder.visibility = View.GONE
            }
            else{
                btnEndOrder.visibility = View.GONE
                btnConfirmOrder.visibility = View.VISIBLE
            }

            btnConfirmOrder.setOnClickListener {
                orderInfo!!.idCourier = courier!!.id
                orderInfo!!.status = "Выдан курьеру"
                ApiService().updateOrderStatus(orderInfo!!){
                    if(it != null){
                        Toast.makeText(applicationContext, "Вы успешно взяли 
заказ", Toast.LENGTH_LONG).show()
                        finish()
                    }
                    else{
                        Toast.makeText(applicationContext, "Неизвестная ошибка, 
повторите позже", Toast.LENGTH_LONG).show()
                    }
                }
            }

            btnEndOrder.setOnClickListener {
                DialogFinishOrderFragment().show(supportFragmentManager, 
"dialog")
            }
        }
        catch (e: Exception){
            tvAddress.text = e.message.toString()
        }

        recyclerProducts.layoutManager = LinearLayoutManager(applicationContext)
        recyclerProducts.adapter =
            OrderComposAdapter(
                (intent.getSerializableExtra("orderCompos") as 
OrderCompos?)!!.listOrderCompos
            )
    }
}
```

4) AccountFragment.kt

```
package com.example.dpstuffprovider.fragments
```

```
import android.content.Context
import android.content.Intent
import android.content.SharedPreferences
import android.os.Bundle
import androidx.fragment.app.Fragment
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import android.widget.FrameLayout
import com.example.dpstuffprovider.MainActivity
import com.example.dpstuffprovider.MainMenu
import com.example.dpstuffprovider.R
import kotlinx.android.synthetic.main.fragment_account.view.*

/**
 * Фрагмент профиля курьера, где отображается личная информация курьера
 */
class AccountFragment : Fragment() {

    override fun onCreateView(
        inflater: LayoutInflater, container: ViewGroup?,
        savedInstanceState: Bundle?
    ): View? {

        val inflate = inflater.inflate(R.layout.fragment_account, container,
false)
        val mainMenu = activity as MainMenu

        inflate.nameCourier.text = "${mainMenu.courier!!.lastName}
${mainMenu.courier!!.firstName} ${mainMenu.courier!!.patronymic}"
        inflate.tvAccPhone.text = "Телефон: ${mainMenu.courier!!.phone}"
        inflate.tvAccEmail.text = "Почта: ${mainMenu.courier!!.email}"
        inflate.tvAccOrdersQuantity.text = "Выполненных заказов:
${mainMenu.courier!!.orderQuantity}"

        inflate.btnExit.setOnClickListener {
            val parentActivity = activity as MainMenu
            val editor : SharedPreferences.Editor =
parentActivity.getSharedPreferences("sp", Context.MODE_PRIVATE).edit()
            editor.putBoolean("isLogin", false)
            editor.apply()

            val intent : Intent = Intent(requireContext(),
MainActivity::class.java)
            startActivity(intent)
        }

        return inflate
    }
}
```

5) ActiveDeliveryFragment.kt

```
package com.example.dpstuffprovider.fragments

import android.os.Bundle
import androidx.fragment.app.Fragment
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import android.widget.FrameLayout
import androidx.recyclerview.widget.LinearLayoutManager
import com.example.dpstuffprovider.MainMenu
import com.example.dpstuffprovider.R
import com.example.dpstuffprovider.adapters.OrdersAdapter
import com.example.dpstuffprovider.api.ApiService
import kotlinx.android.synthetic.main.fragment_active_delivery.*
import kotlinx.android.synthetic.main.fragment_active_delivery.view.*
import kotlinx.android.synthetic.main.fragment_all_delivery.view.*

/**
 * Фрагмент с активными заказами курьера
 */
class ActiveDeliveryFragment : Fragment() {

    override fun onCreateView(
        inflater: LayoutInflater, container: ViewGroup?,
        savedInstanceState: Bundle?
    ): View? {
        val inflate = inflater.inflate(R.layout.fragment_active_delivery,
container, false)
        val parentActivity = activity as MainMenu

        inflate.recyclerActiveOrders.layoutManager =
LinearLayoutManager(context!!)

        inflate.progressBarActive.visibility = View.VISIBLE
        ApiService().getActiveOrdersByCourier(parentActivity.courier!!.id) {
            if(it != null){
                inflate.progressBarActive.visibility = View.GONE
                inflate.hintEmpty.visibility = View.GONE
                inflate.recyclerActiveOrders.adapter = OrdersAdapter(it,
parentActivity)
            }
            else{
                inflate.progressBarActive.visibility = View.GONE
                inflate.hintEmpty.visibility = View.VISIBLE
            }
        }

        return inflate
    }
}
```

6) AllDeliveryFragment.kt

```
package com.example.dpstuffprovider.fragments

import android.os.Bundle
import androidx.fragment.app.Fragment
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import android.widget.FrameLayout
import android.widget.Toast
import androidx.recyclerview.widget.LinearLayoutManager
import com.example.dpstuffprovider.MainMenu
import com.example.dpstuffprovider.interfaces.IOrders
import com.example.dpstuffprovider.adapters.OrdersAdapter
import com.example.dpstuffprovider.R
import com.example.dpstuffprovider.api.ApiService
import com.example.dpstuffprovider.models.OrdersData
import kotlinx.android.synthetic.main.fragment_all_delivery.*
import kotlinx.android.synthetic.main.fragment_all_delivery.view.*
import kotlinx.coroutines.*
import retrofit2.Call
import retrofit2.Callback
import retrofit2.Response
import retrofit2.Retrofit
import retrofit2.converter.gson.GsonConverterFactory
```

```
/**
 * Фрагмент со списком доступных заказов
 */
class AllDeliveryFragment : Fragment() {

    override fun onCreateView(
        inflater: LayoutInflater, container: ViewGroup?,
        savedInstanceState: Bundle?
    ): View? {

        val inflate = inflater.inflate(R.layout.fragment_all_delivery,
container, false)
        val parentActivity = activity as MainMenu

        inflate.recyclerOrders.layoutManager = LinearLayoutManager(context!!)

        inflate.progressBarAll.visibility = View.VISIBLE
        ApiService().getOrdersByStatus("Обрабатывается") {
            if(it != null){
                inflate.progressBarAll.visibility = View.GONE
                inflate.recyclerOrders.adapter = OrdersAdapter(it,
parentActivity)
            }
            else{
                inflate.progressBarAll.visibility = View.GONE
                //Обработка ошибки
            }
        }

        return inflate;
    }
}
```

7) DialogFinishOrderFragment.kt

```
package com.example.dpstuffprovider.fragments

import android.os.Bundle
import android.util.Log
import androidx.fragment.app.Fragment
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import android.view.inputmethod.EditorInfo
import android.widget.Toast
import androidx.fragment.app.DialogFragment
import com.example.dpstuffprovider.MoreInfo
import com.example.dpstuffprovider.R
import com.example.dpstuffprovider.api.ApiService
import kotlinx.android.synthetic.main.fragment_dialog_finish_order.view.*

class DialogFinishOrderFragment : DialogFragment() {

    override fun onCreateView(
        inflater: LayoutInflater, container: ViewGroup?,
        savedInstanceState: Bundle?
    ): View? {
        val inflate = inflater.inflate(R.layout.fragment_dialog_finish_order,
container, false)
        val parentActivity = activity as MoreInfo

        inflate.inputTextCode.setOnEditorActionListener { v, actionId, event ->
            if(actionId == EditorInfo.IME_ACTION_DONE){
                if(inflate.inputTextCode.text.toString() ==
parentActivity.orderInfo!!.codeToFinish){
                    Log.i("myLog", "Код верен")
                    parentActivity.orderInfo!!.status = "Завершен"
                    ApiService().updateOrderStatus(parentActivity.orderInfo!!){
                        if(it != null){
                            parentActivity.courier!!.orderQuantity++

ApiService().updateOrderQuantity(parentActivity.courier!!){
                                if(it != null){
                                    Toast.makeText(requireContext(), "Заказ
завершен", Toast.LENGTH_LONG).show()
                                    this@DialogFinishOrderFragment.dismiss()
                                    parentActivity.finish()
                                }
                                else{
                                    parentActivity.courier!!.orderQuantity--
                                    Toast.makeText(requireContext(),
"Неизвестная ошибка, повторите позже", Toast.LENGTH_LONG).show()
                                }
                            }
                        }
                        else{
                            Toast.makeText(requireContext(), "Неизвестная
ошибка, повторите позже", Toast.LENGTH_LONG).show()
                        }
                    }
                }
                else{
                    Log.i("myLog", "Код не верен")
                    Toast.makeText(requireContext(), "Неверный код",
Toast.LENGTH_LONG).show()
                }

                true
            } else {
                false
            }
        }

        return inflate
    }

}
```

8) OrderComposAdapter.kt

```
package com.example.dpstuffprovider.adapters

import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import android.widget.TextView
import androidx.annotation.NonNull
import androidx.recyclerview.widget.RecyclerView
import com.example.dpstuffprovider.R
import com.example.dpstuffprovider.models.OrderComposData

/**
 * Адаптер для отображения состава заказа (список товаров, входящих в заказ)
 */
internal class OrderComposAdapter(private val orderComposList:
List<OrderComposData>) : RecyclerView.Adapter<OrderComposAdapter.MyViewHolder>()
{
    internal class MyViewHolder(view: View) : RecyclerView.ViewHolder(view){
        val title : TextView = view.findViewById(R.id.tvTitle)
        val storeName : TextView = view.findViewById(R.id.tvStore)
        val quantity : TextView = view.findViewById(R.id.tvQuantity)
    }

    @NonNull
    override fun onCreateViewHolder(parent: ViewGroup, viewType: Int):
MyViewHolder {
        val itemView = LayoutInflater.from(parent.context)
            .inflate(R.layout.order_products_item, parent, false)
        return MyViewHolder(
            itemView
        )
    }

    override fun onBindViewHolder(holder: MyViewHolder, position: Int) {
        holder.title.text = orderComposList[position].name
        holder.storeName.text = orderComposList[position].shopName
        holder.quantity.text = "${orderComposList[position].quantity} Шт"
    }

    override fun getItemCount(): Int {
        return orderComposList.size
    }
}
```

9) OrdersAdapter.kt

```
package com.example.dpstuffprovider.adapters

import android.content.Intent
import android.os.Bundle
import android.util.Log
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import android.widget.Button
import android.widget.TextView
import androidx.annotation.NonNull
import androidx.core.content.ContextCompat.startActivity
import androidx.recyclerview.widget.RecyclerView
import com.example.dpstuffprovider.MainMenu
import com.example.dpstuffprovider.interfaces.IOrders
import com.example.dpstuffprovider.MoreInfo
import com.example.dpstuffprovider.R
import com.example.dpstuffprovider.api.ApiService
import com.example.dpstuffprovider.models.ClientData
import com.example.dpstuffprovider.models.OrderCompos
import com.example.dpstuffprovider.models.OrderComposData
import com.example.dpstuffprovider.models.OrdersData
import retrofit2.*
import retrofit2.converter.gson.GsonConverterFactory

/**
 * Адаптер для отображения списка заказов
 */
internal class OrdersAdapter (private var ordersList: List<OrdersData>, private
var activity: MainMenu) : RecyclerView.Adapter<OrdersAdapter.MyViewHolder>(){
    internal class MyViewHolder(view: View) : RecyclerView.ViewHolder(view){
        val address : TextView = view.findViewById(R.id.tvAddress)
        val date : TextView = view.findViewById(R.id.tvDate)
        val timeFrom : TextView = view.findViewById(R.id.tvTimeFrom)
        val timeTo : TextView = view.findViewById(R.id.tvTimeTo)
        val moreInfo : Button = view.findViewById(R.id.btnMoreInfo)
        //val orderInfo : OrdersData =
    }

    @NonNull
    override fun onCreateViewHolder(parent: ViewGroup, viewType: Int):
MyViewHolder {
        val itemView = LayoutInflater.from(parent.context)
            .inflate(R.layout.order_item, parent, false)
        return MyViewHolder(
            itemView
        )
    }

    override fun onBindViewHolder(holder: MyViewHolder, position: Int) {
        Log.e("", "onBindViewHolder started")
        holder.address.text = ordersList[position].address
        holder.date.text = ordersList[position].deliveryDate
        holder.timeFrom.text = "С ${ordersList[position].timeFrom}"
        holder.timeTo.text = "До ${ordersList[position].timeTo}"


        holder.moreInfo.setOnClickListener{

            val intent : Intent = Intent(holder.itemView.context,
MoreInfo::class.java)

            ApiService().getClient(ordersList[position].id) {
                if(it != null){
                    intent.putExtra("orderInfo", ordersList[position])
                    intent.putExtra("clientInfo", it[0])

                    ApiService().getOrderCompos(ordersList[position].id) {
                        if(it != null) {
                            intent.putExtra("orderCompos", OrderCompos(it))
                            intent.putExtra("courier", activity.courier)
                            startActivity(holder.itemView.context, intent,
Bundle.EMPTY)
                        }
                    }
                }
            }
        }
    }

    override fun getItemCount(): Int {
        return ordersList.size
    }
}
```