

ОТБОРНА ЗАДАЧА

RAPPING RABBIT

FOOD FORMULA

23:13

Food Formula

wine

Strawberry Rosé Wine Punch

Parmesan Cheese Spread

Elderflower Wine Cocktails

Cotton Candy Cherry Lime Champagne Cocktail

Chicken Bone Broth

Honeydew Mimosas

Easy Balsamic-Glazed Filet Mignon

Champagne Cocktail

3 Ingredient Mango Champagne Slushies

Celery Margarita

23:12

Food Formula

 raspberry jam 0.3333333333333333

 vanilla 0.5

 water 0.5

 yeast 2.25


Instruction

In a measuring cup, stir warm water, yeast and 1 tablespoon of sugar together. Set aside to proof. In a large bowl, stir 2 cups of pancake mix together with 1/4 cup granulated sugar and melted butter. Pour in yeast mixture and stir well to get rid of most of the lumps. Stir in remaining 1/2 cup pancake mix. Pour into greased bowl, cover and refrigerate 12 hours or overnight. In the morning, roll out chilled dough on well floured board into a square slightly larger than 12x12. Cut off uneven edges to get a 12x12 even square. Spread jam over dough entirely, then sprinkle with sugar and cinnamon. Roll into log, cut into 6 2-inch pieces and place into parchment paper lined jumbo muffin tin. Preheat oven to 375 degrees. Place rolls next to the oven to come to room temperature quicker. Bake 15-20 minutes or until edges start to brown slightly. Remove from oven. For the glaze, stir all ingredients together until your desired consistency.

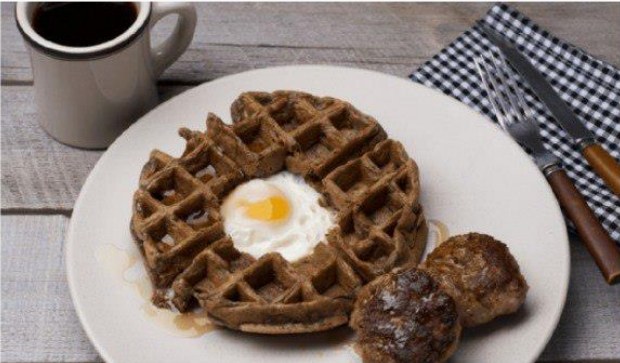
23:33

Food Formula


Toad In The Waffle


 172


Time for preparing in minutes: 50





Ingridients

 baking powder 1.5

 baking soda 0.5

 buckwheat flour 0.5


 butter 2.0

 buttermilk 1.0


23:13

Food Formula


Cotton Candy Cherry Lime Champagne Cocktail


 6


Time for preparing in minutes: 45



Ingridients

 champagne 3.0

 ice 1.0

 m&m candy 1.0

Instruction

Fill champagne flutes with champagne and Sparkling Ice Cherry Limeade. Place a small 2" piece of cotton candy

PROJECT OWNER

PRODUCT ROADMAP

[illegible]

PROJECT MANAGER

PRODUCT BACKLOG

Backlog

...

Добавяне на контент и функционалност, свързани с храна

Да се направи UML диаграма на отделните Use cases на проекта

Да се разработи Activity диаграма

Да се разработи Class / Component диаграма

Да се разработи схема на отделните Database обекти

+ Добавить еще одну карточку

Doing

...

Добавяне на категории

☑ 0/3

+ Добавить еще одну карточку

in Testing

...

Въвеждане на критерии за търсене

☑ 2/3

+ Добавить еще одну карточку

Done

...

Конфигуриране на връзка със сървър

Получаване на данни от сървър

Показване на получените данни

App Bar

Показване на изображението на алкохол

Показване на изображението на храна

Търсене по категория

+ Добавить еще одну карточку

2.1 Оценка на риска

Основните рискове които могат да настъпят при работа с приложението.

#	Риск	Impact	Предизвиква се от	Как може да се справим с него
1	Сигурност	Висок	Опит за злонамерено влизане в чужд профил	Тестване на нивото на сигурност с по сложни пароли и прочие
2	Сигурност	Среден	Опит за редактиране или изтриване на данни на други потребители	Уведомяване на потребителя, че неговите данни са били редактирани или искане за разрешение
3	Достоверност	Висок	Опит за преднамерено добавяне на зловредни данни	Добавяне на забранени данни в БД и автоматична проверка за тяхната наличност при публикуване
4	Използваемост	Среден	Бавна работа на приложението или неправилна визуализация	Тестване на използваемост на приложение

3 Тестови сценарии

Тестови сценарии, които са необходими за ръчното тестване на даден компонент.

Тест на Home screen

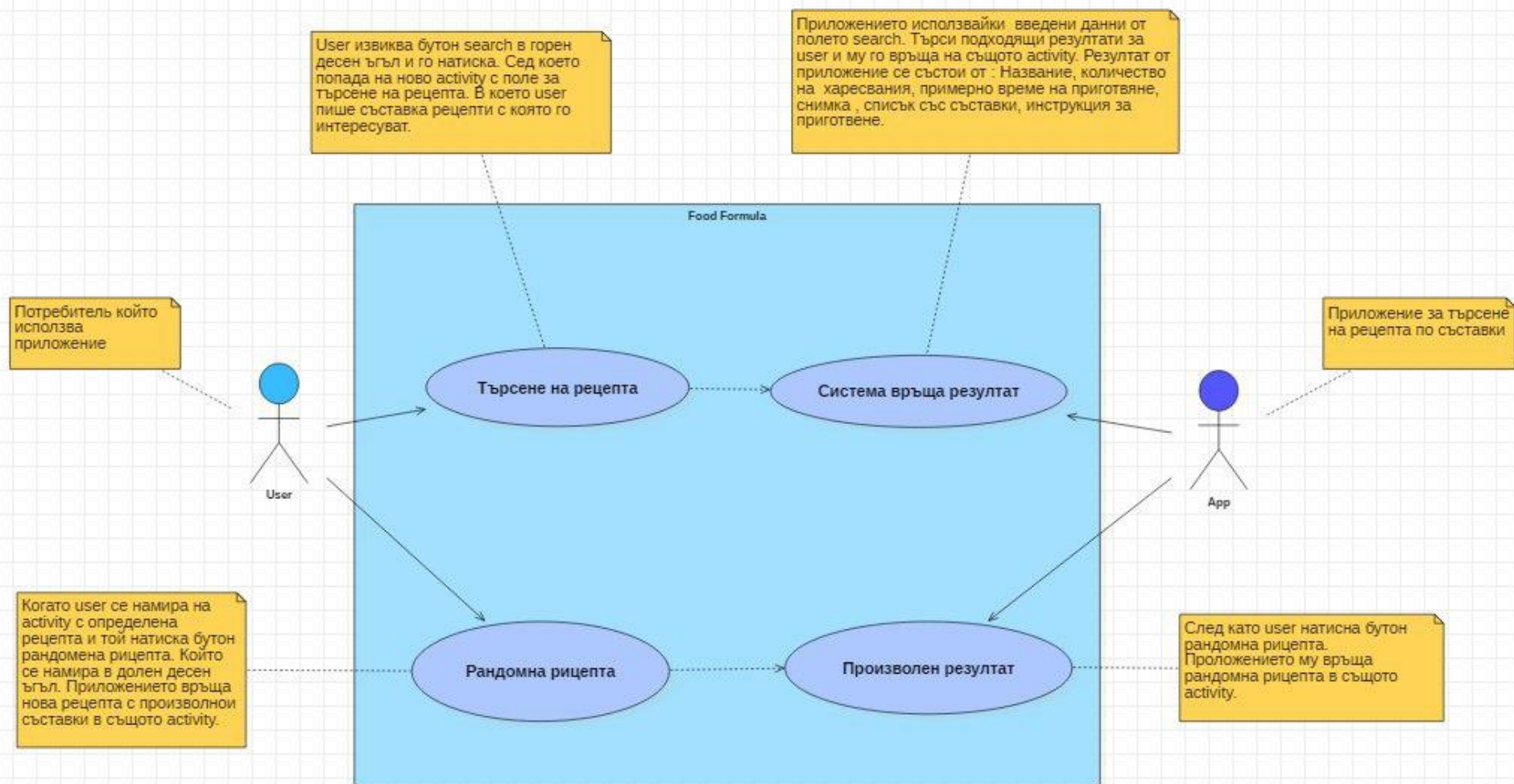
- На главната страница намира се списък на коктейли и ястия.
- След щракване върху коктейл трябва да се отвори страница със списък на съставките на коктейлите, рецепта и снимка му.
- След щракване върху ястие трябва да се отвори страница със списък на съставките на ястие, рецепта и снимка му.
- Има функция за търсене по съставки.
- Също доступна функция за избиране на случаен коктейл или ястие.

Тест на Search screen

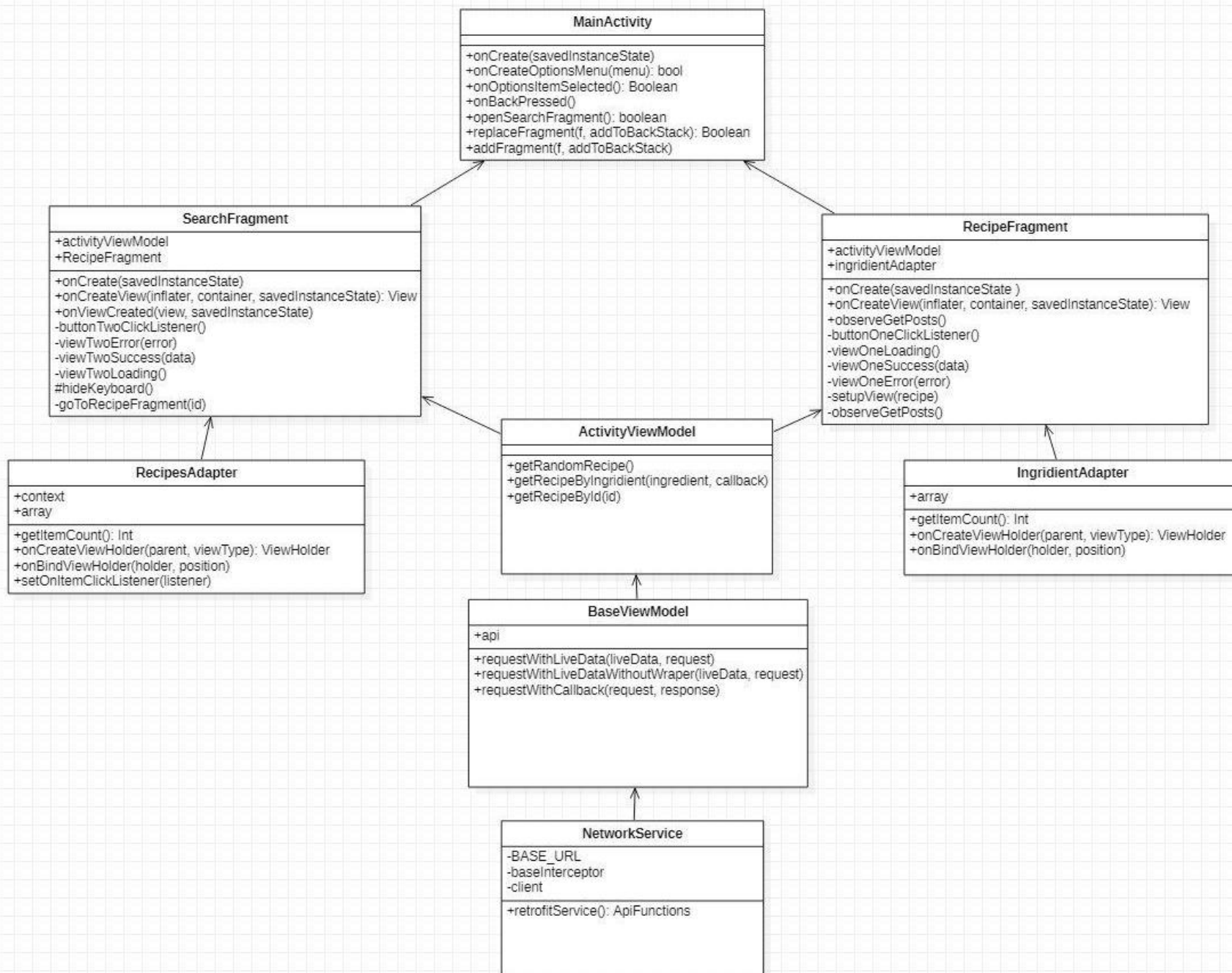
- Потребителят търси коктейли по съставка.
- Очакван резултат - Откриване на страница с всички подходящи коктейли.
- След щракване върху коктейл, той трябва да се появи на главен екран.
- Потребителят търси ястия по съставка.
- Очакван резултат - Откриване на страница с всички подходящи ястия.
- След щракване върху ястие, тя трябва да се появи на главен екран.

DEVELOPER 1

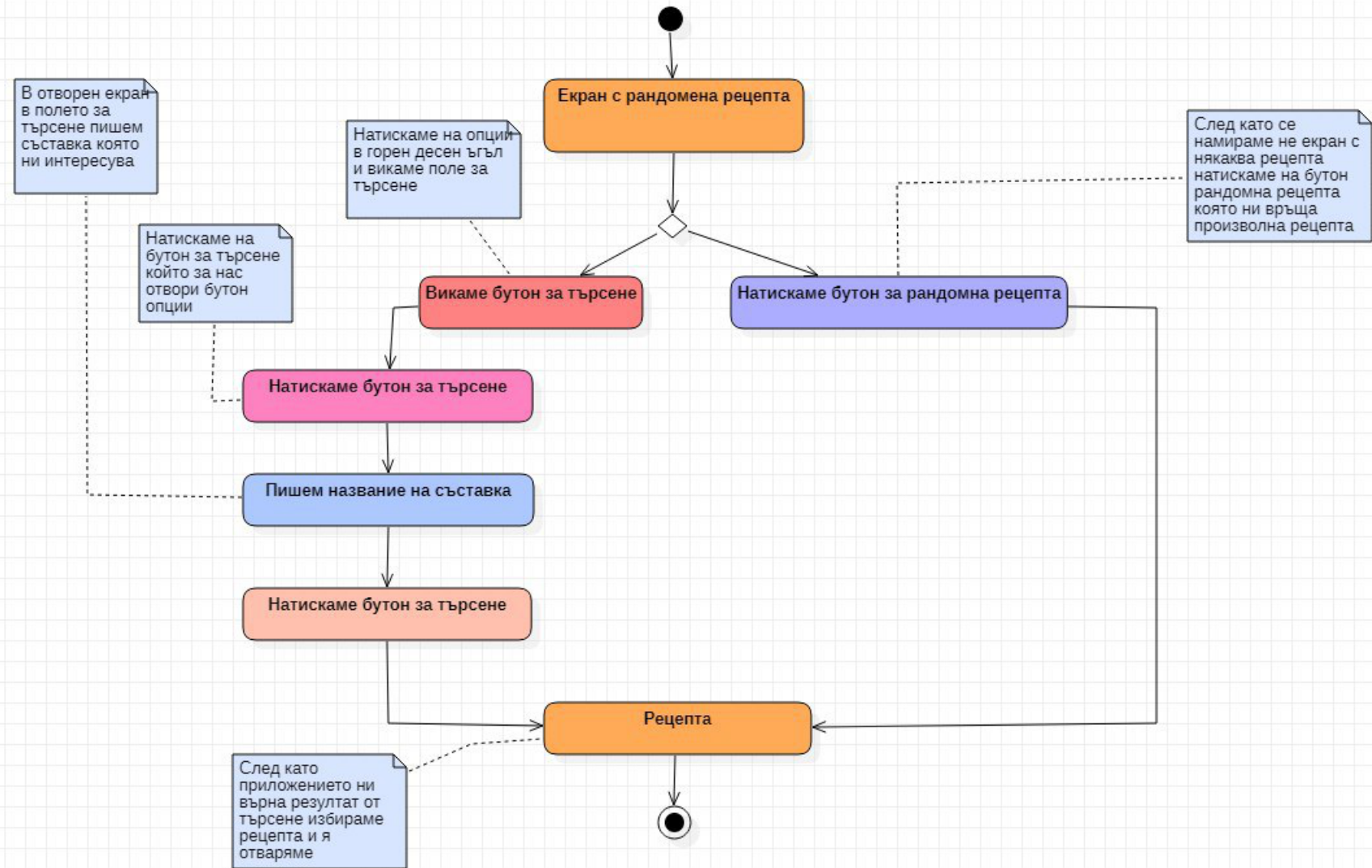
USE CASE DIAGRAM



CLASS DIAGRAM



ACTIVITY DIAGRAM



DEVELOPER 2

```
package com.example.foodformula.UI

import ...

class RecipeFragment : Fragment() {

    companion object {
        fun newInstance(): RecipeFragment = RecipeFragment()
        const val RECIPE_FRAGMENT_KEY = "RECIPE_FRAGMENT_KEY"
    }

    private lateinit var activityViewModel: ActivityViewModel
    var progressDialog: ProgressDialog? = null
    var ingredientAdapter: IngredientAdapter? = null

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)

        activityViewModel = ViewModelProviders.of(fragment = this).get(ActivityViewModel::class.java)
        observeGetPosts()
        if (arguments != null) {
            if (arguments!!.containsKey(SearchFragment.RECIPE_KEY)) activityViewModel.getRecipeById(
                arguments!!.getInt(SearchFragment.RECIPE_KEY)
            )
            else activityViewModel.getRandomRecipe()
        }
    }

    override fun onCreateView(inflater: LayoutInflater, container: ViewGroup?, savedInstanceState: Bundle?) {
        super.onCreateView(inflater, container, savedInstanceState)
        buttonOneClickListener()
        ingredientAdapter = IngredientAdapter(arrayListOf())
        val llm3 = LinearLayoutManager(context)
        llm3.orientation = LinearLayoutManager.VERTICAL
        extendedIngredients.layoutManager = llm3
        extendedIngredients.isNestedScrollingEnabled = false
    }

    private fun observeGetPosts() = activityViewModel.simpleLiveData.observe(ownenr this, Observer {
        when (it.status) {
            Status.LOADING -> viewOneLoading()
            Status.SUCCESS -> viewOneSuccess(it.data)
            Status.ERROR -> viewOneError(it.error)
        }
    })
}
```

```
package com.example.foodformula

import ...

class MainActivity : AppCompatActivity() {

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        Fresco.initialize(context = this)
        setContentView(R.layout.activity_main)
        setSupportActionBar(toolbar)
        addFragment(RecipeFragment.newInstance(), RecipeFragment.RECIPE_FRAGMENT_KEY)
    }

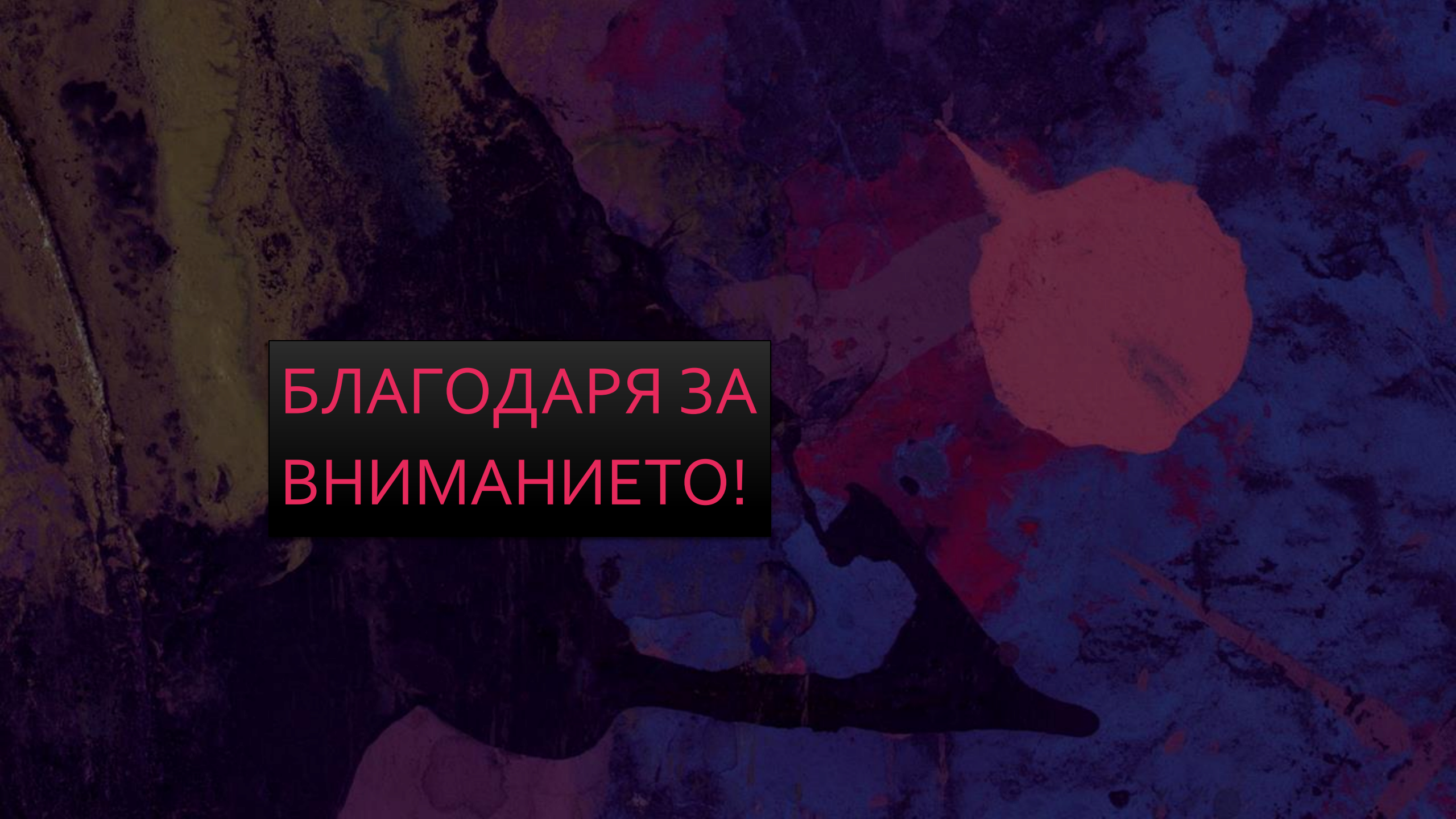
    override fun onCreateOptionsMenu(menu: Menu?): Boolean {
        menuInflater.inflate(R.menu.menu_main, menu)
        return super.onCreateOptionsMenu(menu)
    }

    override fun onOptionsItemSelected(item: MenuItem): Boolean = when (item.itemId) {
        R.id.action_search -> openSearchFragment()
        else -> super.onOptionsItemSelected(item)
    }

    override fun onBackPressed() {
        val currentFragment =
            supportFragmentManager.findFragmentByTag(SearchFragment.RECIPE_KEY) as SearchFragment?
        if (currentFragment != null && currentFragment.isVisible) {
            replaceFragment(
                RecipeFragment.newInstance(),
                RecipeFragment.RECIPE_FRAGMENT_KEY,
                Bundle()
            )
        } else finish()
    }

    private fun openSearchFragment(): Boolean =
        replaceFragment(SearchFragment.newInstance(), SearchFragment.RECIPE_KEY, Bundle())

    private fun addFragment(f: Fragment, addToBackStack: String?) {
        val bundle = Bundle()
        val transaction = supportFragmentManager.beginTransaction()
        transaction.add(R.id.fragment_holder, f, addToBackStack)
        f.arguments = bundle
        transaction.commit()
    }
}
```

БЛАГОДАРЯ ЗА
ВНИМАНИЕТО!