Отборна задача

Rapping rabbit

"Densite 789" е приложение, където можете да получите лесен и бърз достъп до приготвянето на всякакви напитки. Независимо дали става въпрос за лате макиато или любимия ви коледен коктейлимаме всичко. Нашето предимство е, че все още няма нормално и функционално приложение, което да удовлетвори широк спектър от потребителски заявки.





Product owner

Product Roudmap на нашия проект. Проект е пуснат на итерации, като в края на всяка итерациия ще имаме завършена версия, която ще работи.

		Октомври		Ноември			Декември		
	1-10	11-21	21-31	1-10	11-21	21-30	1-10	11-21	21-31
Sketch									
Prototyping									
Design									
Development									
Agreement									
Testing									
Realease									
Marketing									

1 етап: създаване на sketch и prototyping

- Спринт 1: версия 1 схема на база данни (public api)
- Спринт 2: създаването на прототип на приложение (създаване на базов проект и конфигуриране на връзка със сървъра)

2 етап: добавяне на базова функционалност на приложение

- Спринт 3: създаването на функция за случаен коктейл
- Спринт 4: създаването на search на коктейли и на коктейли по съставки

3 етап: добавянето на останалата част от функционалността

- Спринт 5: функция за добавяне на собствен коктейл
- Спринт 6: bug-fixing преди design

4 етап: този етап посвятен на тестването и добавянето на design

- Спринт 7: добавяне на design към приложение
- Спринт 8: тестване на АРІ-то

5 етап: този етап нацелен на релиз на приложението

- Спринт 9: релиз на приложение
- @ маркетингова кампания, насочена към реклама, се разработва паралелно през целия проект

Етапи за разработване на приложение.



Риск план.

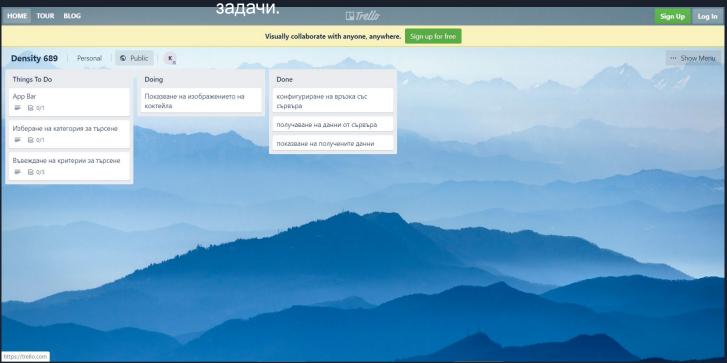
2.1 Оценка на риска

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	Използваемос т	Среден	Бавна работа на приложението или неправилна визуализация	Тестване на исползваемост на приложение			

Product manager

User stories карти, които описват какви са основните



Dev 2

```
fetch(String url) async {
  print("start");
  final JsonDecoder _ decoder = new JsonDecoder();
  Response newResponse = await get(url);
  final String isonBody = newResponse.body;
  final statusCode = newResponse.statusCode;
 if (statusCode < 200 || statusCode >= 300 || jsonBody == null) {
    throw Exception('Failed to load post');
  final recipesContainer = _decoder.convert(jsonBody);
  final Map<String, dynamic> testMap = recipesContainer['drinks'][0];
  final List recipeItems = recipesContainer['drinks'];
if (recipeItems == null) {
    strDrink = "NaN";
    strDrinkThumb = "":
} else {
    strDrink = recipeItems[0]["strDrink"];
    strDrinkThumb = recipeItems[0]["strDrinkThumb"];
    strInstructions = recipeItems[0]["strInstructions"];
    strIngredient.clear();
    for (int i = 0; i < testMap.length; ++i) {</pre>
      if (testMap.keys.elementAt(i).toString().startsWith("strIngredient") &&
          testMap.values.elementAt(i).toString().isNotEmpty &&
          testMap.values.elementAt(i).toString() != "null")
        strIngredient.add(testMap.values.elementAt(i));
  print(testMap.toString());
```

```
@override
Widget build(BuildContext context) {
  return Scaffold(
  - body: Center(
    — child: myLayoutWidget()) , // Center
fetchR() async {
  print("start");
  String url = "https://www.thecocktaildb.com/api/ison/v1/1/random.php";
  final JsonDecoder decoder = new JsonDecoder():
  Response newResponse = await get(url):
  final String isonBody = newResponse.body:
  final statusCode = newResponse.statusCode;
  if (statusCode < 200 || statusCode >= 300 || jsonBody == null) {
    throw Exception('Failed to load post');
  final recipesContainer = _decoder.convert(jsonBody);
  final Map<String, dynamic> testMap = recipesContainer['drinks'][0];
  final List recipeItems = recipesContainer['drinks'];
  if (recipeItems == null) {
    strDrink = "NaN":
   strDrinkThumb = "":
  } else {
    strDrink = recipeItems[0]["strDrink"];
    strDrinkThumb = recipeItems[0]["strDrinkThumb"];
    strInstructions = recipeItems[0]["strInstructions"];
    strIngredient.clear();
    for (int i = 0; i < testMap.length; ++i) {</pre>
      if (testMap.keys.elementAt(i).toString().startsWith("strIngredient") &&
          testMap.values.elementAt(i).toString().isNotEmpty &&
          testMap.values.elementAt(i).toString() != "null")
        strIngredient.add(testMap.values.elementAt(i));
  setState(() => strDrink):
  print(testMap.toString()):
Widget myLayoutWidget() {
  return Column(
    mainAxisAlignment: MainAxisAlignment.center,
    children: [
     - Text(' Name : $strDrink'),
     - Image.network(strDrinkThumb),
      Text('Instruction $strInstructions').
      FlatButton(
      -child: Text('Refresh'),
        onPressed: fetchR,
```

Dev 1

