EventMate

Plan-Driven and Agile Programming

TEI Crete, Winter semester 2018/2019

Petra Cendelínová
Petr Kalas

1. System specification

1.1. Introduction

For human beings, it is natural to socialize and interact with each other. Technology has become more and more important in the social aspects of life. Nowadays people prefer to spend more time on portable devices. Social networking has moved to mobile platforms, which are accessible anywhere and anytime.

The objective of our work is to develop a social application for organizing events any type. As events can be considered birthday parties, new year eve, weddings, baby showers etc.
As we want to make this application more general for any kind of events, we have decided to call it by name “EventMate”. The future user is allowed to choose from predefined event types with a corresponding graphics interface. The main goal of the application is to provide tasks management (create, assign, close) and also to create a communication channel among event owners and guests. The application is going to content gamification elements such as a scoreboard.

1.2. Description

The following chapter explains major features. To use the application it is necessary to create a user's account or log in via existing social accounts. In our application, there are various user roles such as owners, contributors, guests. As we mentioned earlier, the main goal is to provide task management for an event. For all practical purposes, it means that user can create a task with corresponding attributes such as name, deadline, and persons to be assigned to it. All these fields are saved and continuously maintained. During the whole event, gamification principles are applied which help users to feel that they are a part of the game. When the event has finished, event summary is provided to particular users.

1.3. Requirements

There are individual features listed:

- Create a user account
- Authorization via Google / Facebook Account
- Various user roles in an event (owners, contributors, target user...)

- Event creation
- Task management including Location-based task
- Persistence of images or videos in order to create an event summary
- Event timeline
- Event score-board (gamification)
- Use of phone camera
- Communication channel
- Notification system (for example, notification for a task which is set to a specific time, notification for task creator when task assignee has submitted result)
- Edit submitted photos by various filters
- Event summary (sending via email)

1.3.1. Requirements list

1.3.1.1. Functional requirements

1.3.1.2. Non-functional requirements

1.3.2. Requirements diagram

asaddas

1.4. Use Case

asdas

1.4.1. Use case diagram

Use case diagram is located at attachment 1.

1.4.2. Use case details

asdasd

1.5. Mockups

asda

1.5.1. Event owner (Android)

asd

1.5.2. Event owner (web)

asd

1.5.3. Event assignee

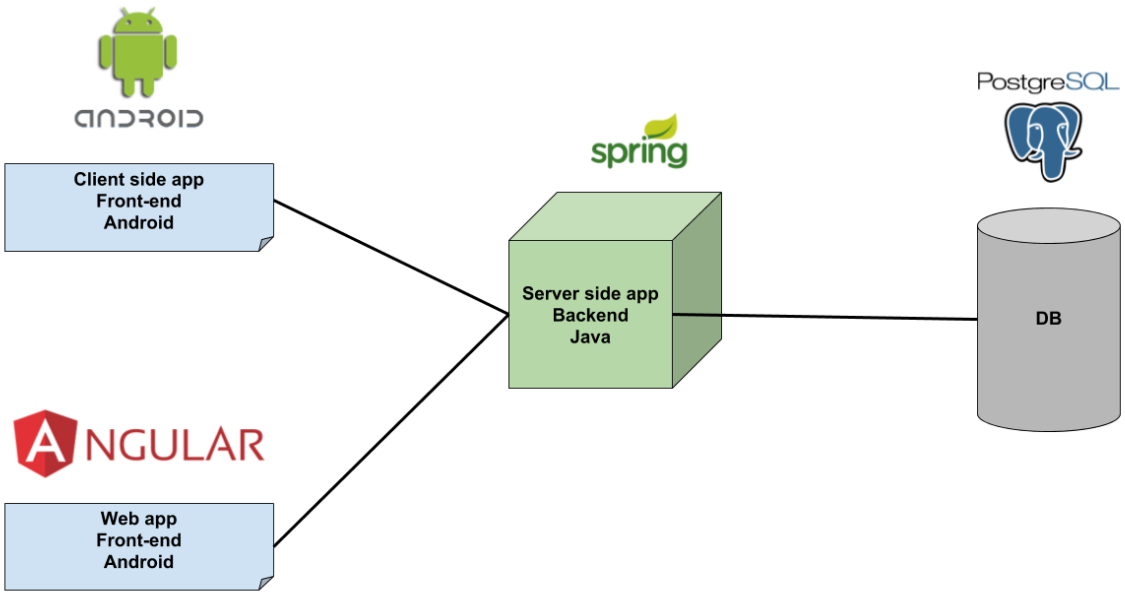
asd

2. Architecture design

Architecture style is based on **client-server.** The project consists of two client side parts and server side with a database.

2.1. Component diagram

2.2. Architecture diagram



Server side

Purpose of server-side application is to provide API, user authentication and persistence to the client side. The asynchronous notification system will be also provided by the server side.

Technology stack (early version)

- Java
- Spring Framework
- PostgreSQL

Client side

Client side - Android

First type of client-side is going to be implemented as a native mobile application for platform Android There will be a huge emphasis on UX (User experience) and also on gamification techniques. Android application is going to adhere MVVM architecture that allows separating the user interface logic and the business logic.

Technology stack

- Kotlin

- LiveData
- Library Retrofit for API calls
- Dagger for dependency injection

Client side – Web

Another type of client-side is going to by implemented as a web application based on Angular framework.

Technology stack
- Angular 6 framework

TODO:

Chapter 1 – system specification (diagrams..)
Description, requeiremnts list + diagram, textual description, use case diagram, mockups, short description,

Chapter 2 (Architecture design (title))
- Component diagram
  - Architecture diagram

Prezentace

1. Slide – mockup
2. Slide – req. diagram
3. Slide – use case diagram

3. Attachments

3.1. Attachment 1 – Use case diagram