



**FAKULTA  
INFORMAČNÍCH  
TECHNOLOGIÍ  
ČVUT V PRAZE**

# **Event Planner**

## **Analysis**

Documentation of a project for the purpose of the course BIE-SI1.

Authors:



## Contents

1. Specification	4
2. Business Process Model	5
2.1 Event Guest	5
2.1.1 Accept	6
2.1.2 Check Invitation	6
2.1.3 Participate	6
2.1.4 Pay	6
2.1.5 Reject	6
2.1.6 Wait for event	6
2.2 Organizer	6
2.2.1 Collect payment	6
2.2.2 Create event	7
2.2.3 Event	7
2.2.4 Invite	7
2.3 System	7
2.3.1 Notify	7
2.3.2 Check deadline	7
3. Requirements	8
3.1 Functional Requirements	8
3.1.1 Account Management	8
3.1.2 Event Management	8
3.1.3 Location management	8
3.1.4 Notification Management	9
3.1.5 Payment Management	9
3.2 Non-Functional Requirements	9
3.2.1 Backup	9
3.2.2 Color scheme of app	9
3.2.3 Localization	9
3.2.4 Web application	10
4. Use Case Model	10
4.1 Accept / Reject	10
4.2 Accept/Reject invitation to event	11
4.3 Cancel event	11
4.4 Collect payments	11
4.5 Create event	11
4.6 Deposit / make payment	11
4.7 Edit event	11
4.8 Enter event chat	11
4.9 Invite others	12
4.10 Leave event	12
4.11 Actors	12
4.11.1 Guest user	12
4.11.2 Logged in user	12
4.11.3 Organizer	12
4.11.4 Event guest	12
4.12 Use Cases	13
4.12.1 Create event	13
4.12.2 Suggest event changes	13
5. Domain Model	14
5.1 Event	14
5.2 Friend	14



5.3	Invitation	14
5.4	Location	15
5.5	Money Transfer	15
5.6	Person	15
5.7	User	15



## 1. Specification

Our goal is to create an app to make it easier for people to create, manage and organize events.

How we will make it easier?

We create a web app through which the users will be able to invite their friends and manage the event organization.

This app is a tool that allows users to create dedicated chat and page for their events. On each of your event pages, guests can learn more, RSVP, and interact with organiser and other guests.

The user can register, create or attend an event that he/she was invited to. The organizer can invite a person that is not registered on the webpage, guests will be invited through email and SMS.



## 2. Business Process Model

Two business process models, for guest and organizer.

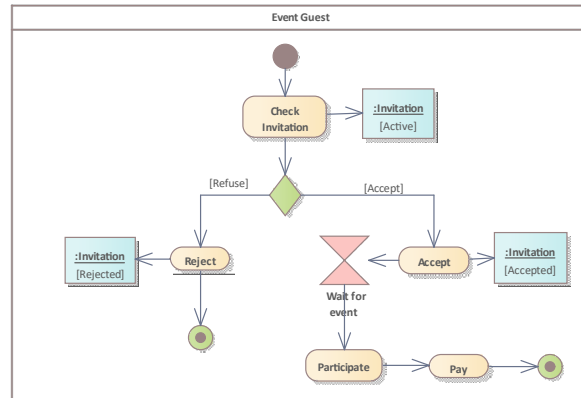


Figure 1 - Guest Diagram

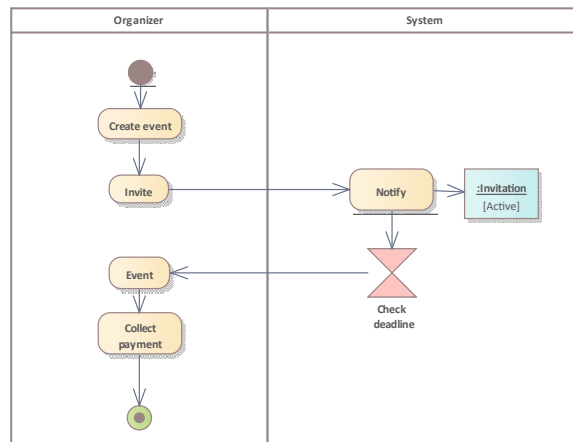


Figure 2 - Organizer Diagram

### 2.1 Event Guest

Receives invitation to event.  
Takes part in event(or does not).  
Pays for participation.



### **2.1.1 Accept**

Invited person accepts invitation and is added to a guest list.

### **2.1.2 Check Invitation**

Invited person decides whether he/she wants to attend an event.

### **2.1.3 Participate**

Guest participates in an event.

### **2.1.4 Pay**

Guest pays for an event.

### **2.1.5 Reject**

Invited person rejects invitation. Does not visit an event and is not added to a guest list.

### **2.1.6 Wait for event**

Guest waits for the start of an event.

## **2.2 Organizer**

Organizes event. Invites people. Holds an event. Collects payments.

### **2.2.1 Collect payment**

Organizer collects payment from guests.



### **2.2.2 Create event**

Organizer creates an event. Gives it name, description, price, capacity, etc.

### **2.2.3 Event**

Organizer holds an event.

### **2.2.4 Invite**

Organizer decides who to invite.

## **2.3 System**

Sends invitations. Checks invitation and event deadlines.

### **2.3.1 Notify**

System notifies(sends email) invited people.

### **2.3.2 Check deadline**

System waits for the start of an event. When event starts, invitations that were not accepted are canceled.



## 3. Requirements

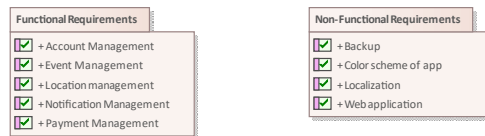


Figure 3 - Requirements

Functional and non-functional requirements diagram.

### 3.1 Functional Requirements

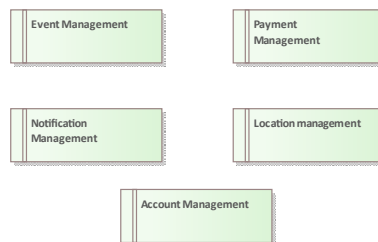


Figure 4 - Functional Requirements

Functional requirements diagram.

#### 3.1.1 Account Management

1. Change account information
2. Change password
3. Add people to friend list

#### 3.1.2 Event Management

1. Keep record/info of event
2. Guest list.

#### 3.1.3 Location management

1. Choose location
2. Change location
3. Add location





### 3.1.4 Notification Management

1. Notifies invited guests
2. Notifies guests about changes
3. Notify who will participate

### 3.1.5 Payment Management

1. Deposit money
2. Mark payments as paid or not
3. List of people who should pay for participation

## 3.2 Non-Functional Requirements



Figure 5 - Non-Functional Requirements

Non-functional requirements diagram.

### 3.2.1 Backup

Save information from time to time.

### 3.2.2 Color scheme of app

Change color scheme of app, according to user preferences.

### 3.2.3 Localization

Application is in several languages.



### 3.2.4 Web application

Can be accessed through web page

## 4. Use Case Model

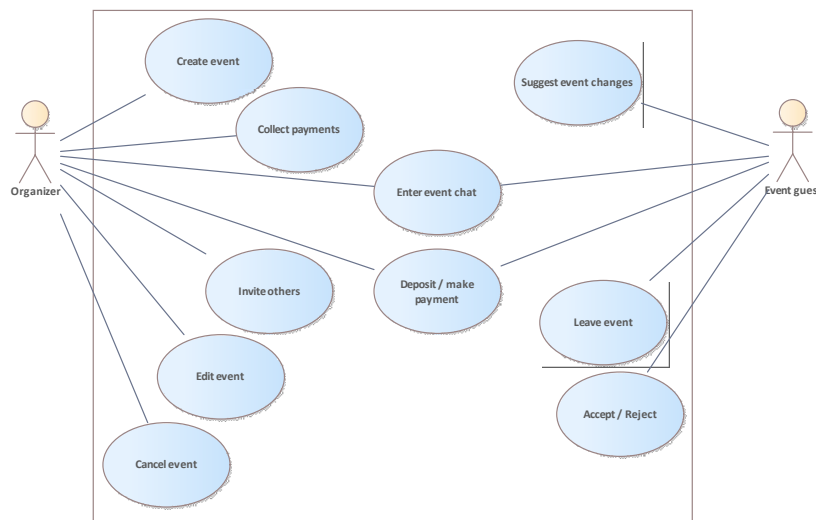


Figure 6 - Use Case Model with Activity Diagram

Use case model of an organizer and an event guest.

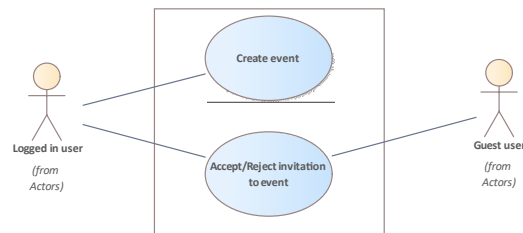


Figure 7 - User Use Case Model

### 4.1 Accept / Reject

Accept or reject invitation to an event.



## **4.2 Accept/Reject invitation to event**

Logged in user or a guest user may be invited to an event. Can accept or reject an invitation.

## **4.3 Cancel event**

Cancels event

## **4.4 Collect payments**

Collects payment from guests. Mark payments as done/not done.

## **4.5 Create event**

Logged in user can create and manage an event.

## **4.6 Deposit / make payment**

Set amount of money person gave to an organizer for future events.

When guest visits and event organizer subtracts from guest deposited money or guest can pay after event if no money was deposited.

## **4.7 Edit event**

Change event details(date, location, time).

## **4.8 Enter event chat**

Use event chat to communicate with an organizer and other guests.



## **4.9 Invite others**

Invites more people to an event.

## **4.10 Leave event**

Do not go to event.

## **4.11 Actors**

Contains Actors that represent the roles that users play with respect to the system.

### **4.11.1 Guest user**

Not registered user. Can only accept invites and participate in event. No event creation for him.

### **4.11.2 Logged in user**

User who created an account and logged in.

### **4.11.3 Organizer**

1. Can create, cancel, and edit event.
2. Responsible for payment management.
3. Can use chat.
4. System notifies about all the changes.
5. Can invite others.

### **4.11.4 Event guest**

1. Can accept/reject invitation.
2. System notifies other event participants.
3. Pays
4. System accepts payments.
5. Can suggest changes / leave event.
6. Is able to use chat.
7. Can request to add someone.



## **4.12 Use Cases**

Contains Use Case that represent the value or goal that the Actors wish to achieve.

### **4.12.1 Create event**

Creates event.  
Add description.  
Choose location.  
Set capacity.  
Invite people.  
Set price.

#### **Basic Path: Basic Path**

1. User Step One
2. System Step One
3. User Step Two
4. System Step Two
5. User Step Three
6. System Step Three

#### **Alternate: Alternate Path A**

1. User Step A One
2. System Step A One

### **4.12.2 Suggest event changes**

Offer changes for event(date, time, location, etc.).

## 5. Domain Model

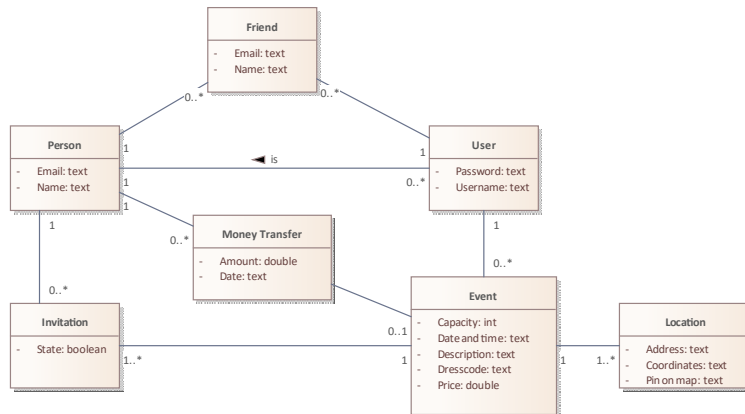


Figure 8 - Domain Model

Domain model of an event planning app.

### 5.1 Event

Event is organized by user. Has a description.

Attribute title	Description
Capacity	How many people can attend an event
Date and time	Date and time of an event
Description	Description of an event
Dresscode	What to wear to an event
Price	Price of whole event

### 5.2 Friend

Each person or user has a friend list, which can be empty.

Attribute title	Description
Email	Email of a friend
Name	Name of a friend

### 5.3 Invitation

Person receives invitation to an event. Can accept or reject.



Attribute title	Description
State	State of invitation(accepted/rejected)

## 5.4 Location

Location of the event. Event may have several locations.

Attribute title	Description
Address	Address of event location
Coordinates	In case event is organized not in town, coordinates might be used
Pin on map	Can pin the location of an event

## 5.5 Money Transfer

Each person can mark down if he paid for the event to an organizer

Attribute title	Description
Amount	Amount guest should pay to an organizer
Date	When guest should pay to an organizer

## 5.6 Person

Person may be invited. Can create a user account and create events.

Attribute title	Description
Email	Email of a person
Name	Name of a person

## 5.7 User

Creates events, invites people, attends events.

Attribute title	Description
Password	User password used to log in
Username	Username used to log in



**FAKULTA  
INFORMAČNÍCH  
TECHNOLOGIÍ  
ČVUT V PRAZE**