FAKULTA
INFORMAČ
TECHNOLO
ČVUT V PR

Event Planner - Copy

Architecture and Database

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

Authors:

FAKULTA
INFORMAČ
TECHNOLO
ČVUT V PR

Contents

1. Architecture
   - 1.1 app
   - 1.2 controller
   - 1.3 data
     - 1.3.1 dao
     - 1.3.2 entity
   - 1.4 exception
   - 1.5 fit.biesp.oneplan
   - 1.6 Package1
   - 1.7 repository
   - 1.8 security
   - 1.9 service
   - 1.10 web
2. System Architecture
3. Database Model
   - 3.1 «table» Event
   - 3.2 «table» Friend
   - 3.3 «table» Invitation
   - 3.4 «table» Location
   - 3.5 «table» Money Transfer
   - 3.6 «table» Person
   - 3.7 «table» User

1. Architecture

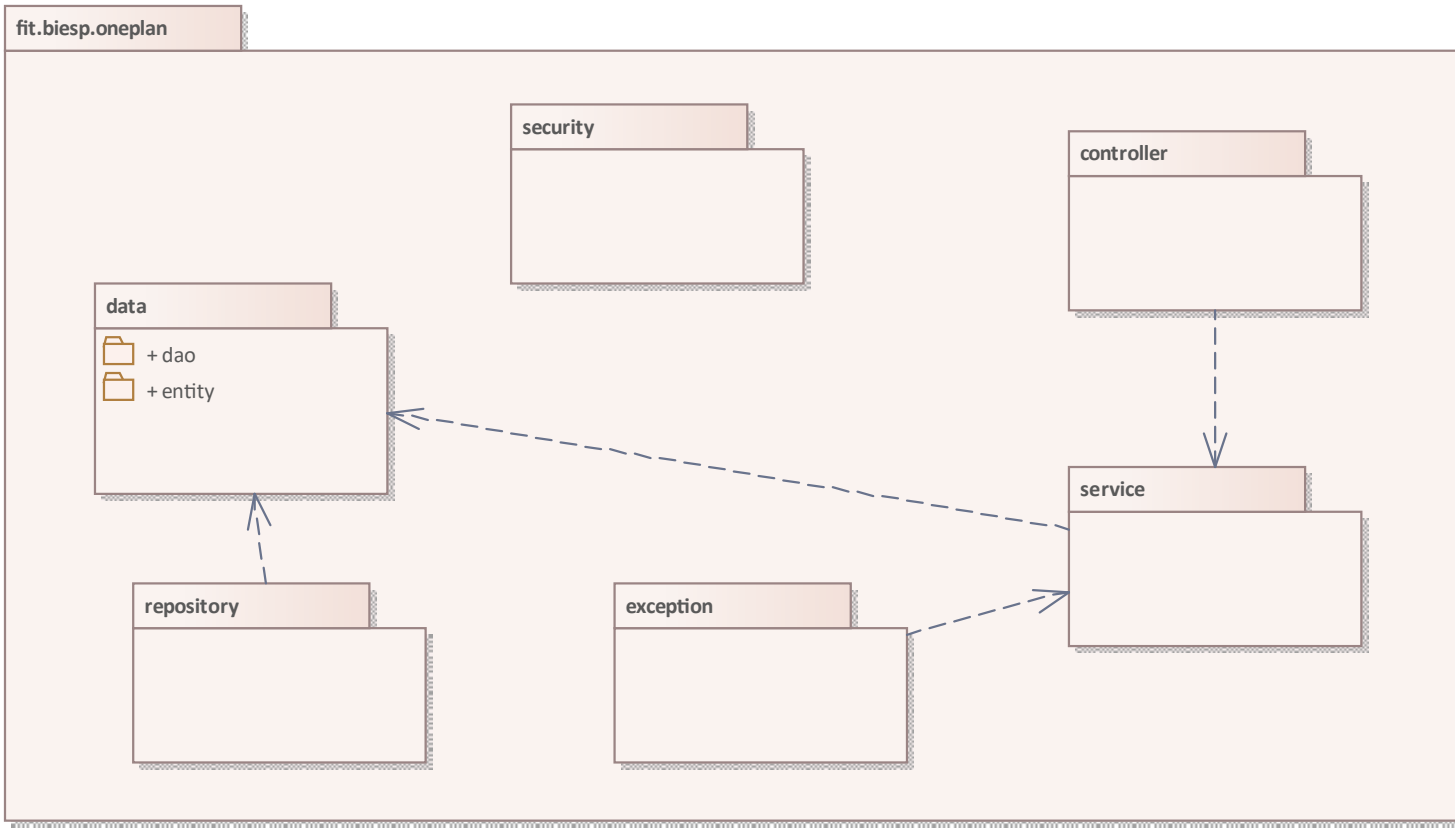

Figure 1 - Architecture

Architecture defines the boundaries between parts of the app and the responsibilities each part should have. There a
data classes, entity is used to store information in the database, while DAO classes are used to transfer information
user and the application. HTTP requests are handled by controller classes. Service classes are used for logic of th
data manipulations. Repository interfaces extend CrudRepository and used for connection with a database. Excep
handled by exception classes.

1.1 app

1.2 controller

Used for HTTP requests handlling.

1.3 data

Data storing classes.

1.3.1 dao

DAO classes used to transfer information between user and the application.

1.3.2 entity

Entities are used to store information in a database.

1.4 exception

Used for exceptions that might occur.

Figure 3 - exception

1.5 fit.biesp.oneplan

Figure 4 - fit.biesp.oneplan

1.6 Package1

1.7 repository

Used for connection with a database.

Figure 5 - repository

1.8 security

Used for secure user log in and registration.

Figure 6 - security

1.9 service

Implementation of app logic and data manipulations.

Figure 7 - service

1.10 web

2. System Architecture

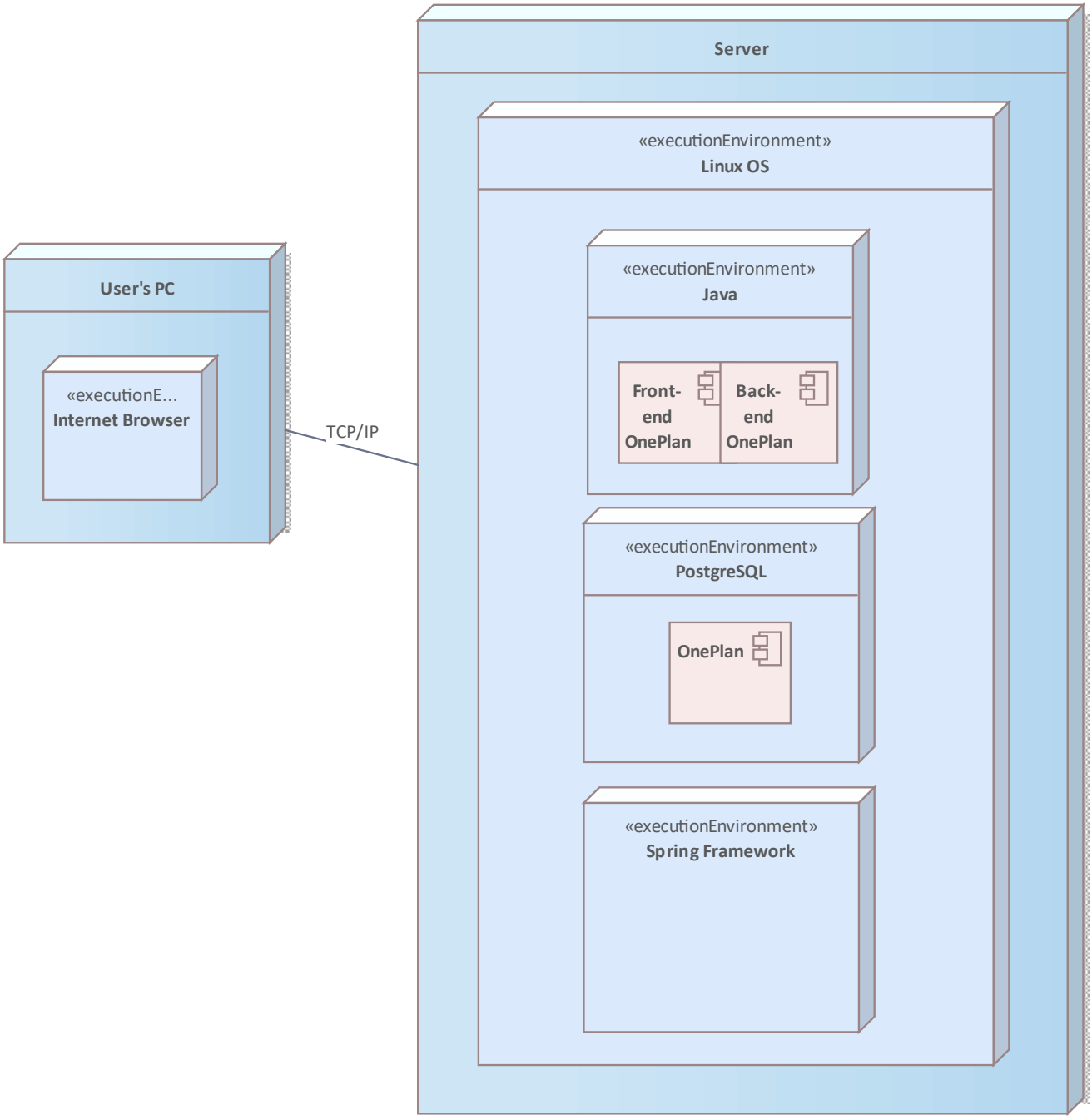

Figure 8 - System Architecture

Conceptual model of system architecture defines the structure and behavior of a system. Server of our app is acces
internet browser on user's PC. Server is running Linux OS, which runs our java plus spring framework application
PostgreSQL database.

FAKULTA
INFORMAČ
TECHNOLO
ČVUT V PR

3. Database Model


Money Transfer
«column»
*FK event_id: integer
* amount: double precision
* date: date
*FK person_id: integer
*PK transfer_id: integer
«FK»
+ FK_Money Transfer_Event_02(integer)
+ FK_Money Transfer_Person_02(integer)
«index»
+ IXFK_Money Transfer_Event(integer)
+ IXFK_Money Transfer_Event_02(integer)
+ IXFK_Money Transfer_Person()
+ IXFK_Money Transfer_Person_02(integer)
«PK»
+ PK_Money Transfer(integer)
Person
«column»
*PK person_id: integer
* person_email: varchar(50)
«PK»
+ PK_Person(integer)
User
«column»
* username: varchar(50)
* password: varchar(50)
*pfK person_id: integer
* person_email: varchar(50)
«FK»
+ FK_User_Event()
+ FK_User_Person_02(integer)
«index»
+ IXFK_User_Friend()
+ IXFK_User_Friend_02()
+ IXFK_User_Person()
+ IXFK_User_Person_02()
+ IXFK_User_Person_03(integer)
+ IXFK_User_Person_04(integer)
«PK»
+ PK_User(integer)
Event
«column»
*FK person_id: integer
* capacity: integer
* description: varchar(10
* price: double precision
* date: date
*PK event_id: integer
* time: time
FK location_id: integer
«FK»
+ FK_Event_Location(int
+ FK_Event_User_02(int
«index»
+ IXFK_Event_Location(i
+ IXFK_Event_User(integ
+ IXFK_Event_User_02(i
«PK»
+ PK_Event(integer)
Friend
«column»
* nickname: varchar(50)
* person_email: varchar(50)
*pfK person_id: integer
«FK»
+ FK_Friend_Person(integer)
+ FK_Friend_User(integer)
«index»
+ IXFK_Friend_Person(varchar)
+ IXFK_Friend_Person_02(integer)
+ IXFK_Friend_User(integer)
«PK»
+ PK_Friend(integer)
Invitation
«column»
*FK event_id: integer
* state: boolean
* user_id: integer
*FK person_id: integer
*PK invitation_id: integer
«FK»
+ FK_Invitation_Event_03(integer)
+ FK_Invitation_Person_02(integer)
«index»
+ IXFK_Invitation_Event(integer)
+ IXFK_Invitation_Event_02(integer)
+ IXFK_Invitation_Event_03(integer)
+ IXFK_Invitation_Friend(integer)
+ IXFK_Invitation_Person(integer)
+ IXFK_Invitation_Person_02(integer)
+ IXFK_Invitation_User(integer)
«PK»
+ PK_Invitation(integer)
Location
«column»
* address: varchar(50)
coordinates: varchar
* event_id: integer
*PK location_id: integer
«index»
+ IXFK_Location_Event
+ IXFK_Location_Event
«PK»
+ PK_Location(integer)
(person_id = person_id)
(event_id = event_id)
(location_id = locat
0..*
1
0..1
1..*

Figure 9 - Database Model

A database model determines the logical structure of a database. It shows how data is stored, organized and mar
This is the relational model, which uses a table-based format. Our database consists of seven entities so far. Pers
stores email of a person who is not necessarily registered. User is a registered person and can log in. Friend entity i
user to add friends to friends list and give friend a nickname. Event contains all the information about the event use
Each event has a location, so location entity is used for this purpose. After event is created invitations are sent
invited by user. Money transfer shows who should pay, when person has to do it, how much and what for.

3.1 «table» Event

Event entity stores information about event created by user.

Column name	Data type	Not null	Description
person_id	integer	True	
capacity	integer	True	
description	varchar(10000)	True	
price	double precision	True	
date	date	True	
event_id	integer	True	
time	time	True	
location_id	integer	False	

3.2 «table» Friend

Friend entity stores persons from user's friend list and their nicknames.

Column name	Data type	Not null	Description
nickname	varchar(50)	True	
person_email	varchar(50)	True	
person_id	integer	True	

3.3 «table» Invitation

Invitation entity stores information about event and who it was sent to.

Column name	Data type	Not null	Description
event_id	integer	True	
state	boolean	True	
user_id	integer	True	
person_id	integer	True	
invitation_id	integer	True	

3.4 «table» Location

Location entity stores information about a location of an organized event.

Column name	Data type	Not null	Description
address	varchar(50)	True	
coordinates	varchar(50)	False	
event_id	integer	True	

3.5 «table» Money Transfer

Money transfer entity stores information about payments that have to be done for events.

Column name	Data type	Not null	Description
event_id	integer	True	
amount	double precision	True	
date	date	True	
person_id	integer	True	
transfer_id	integer	True	

3.6 «table» Person

Person entity stores email of not registered people.

Column name	Data type	Not null	Description
person_id	integer	True	
person_email	varchar(50)	True	

3.7 «table» User

User entity stores information about registered user.

Column name	Data type	Not null	Description
username	varchar(50)	True	
password	varchar(50)	True	
person_id	integer	True	
person_email	varchar(50)	True	

FAKULTA
INFORMAČ
TECHNOLO
ČVUT V PR