**Architectural Design**

**High level components and their interaction**

*Database* :

the data layer is responsible for the data storage and retrieval.

It does not implement any application logic. This layer must guarantee ACID properties.

*Application* server :

this layer contains all the application logic of the system.

All the policies, the algorithms and the computation are performed

here. This layer offers a service-oriented interface.

*Android Mobile application* :

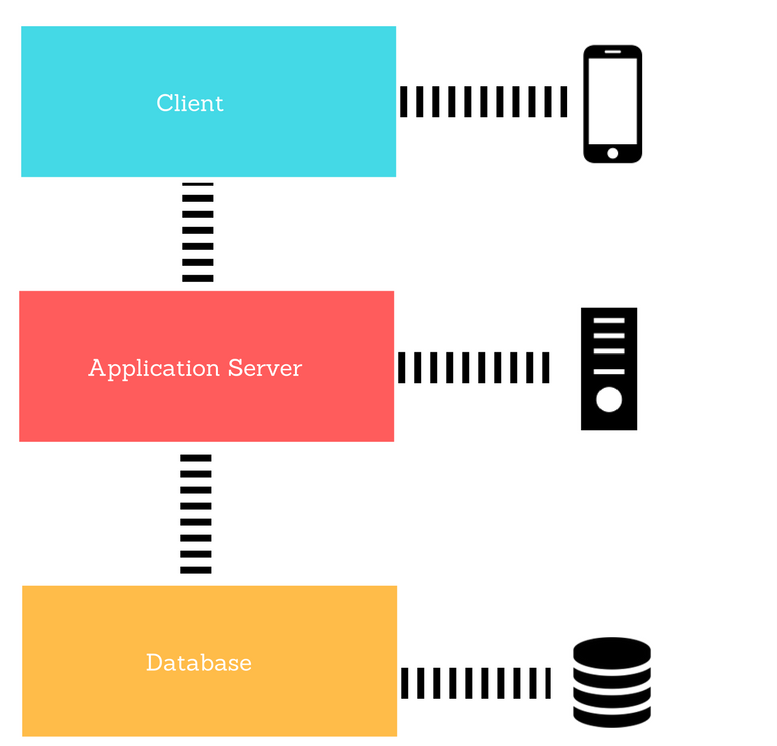
this layer consists in the mobile client. It’s both presentation layer and logic/client layer,

it communicates directly with the application server and it represents the user’s interface.

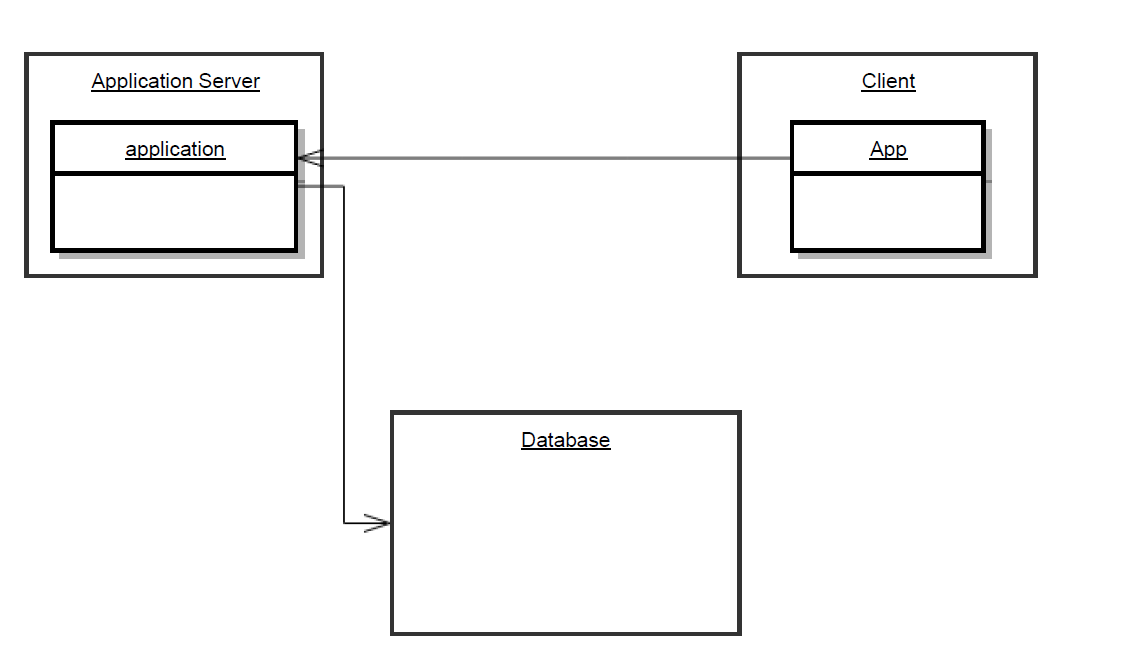
These high-level components are structured into three layers, shown in figure 2.1 .

This choice give us the possibility to compute all the business logic in the Application Server layer, to make more light and efficient to provide a comfortable user experience.

Furthermore this design allows to extend the system, inserting a Web Server layer to consult Travlendar in every device that has a generic browser.



*Figure 2.1: Layers of the system.*



*Figure 2.2: High level components of the system.*

**Component View**

****

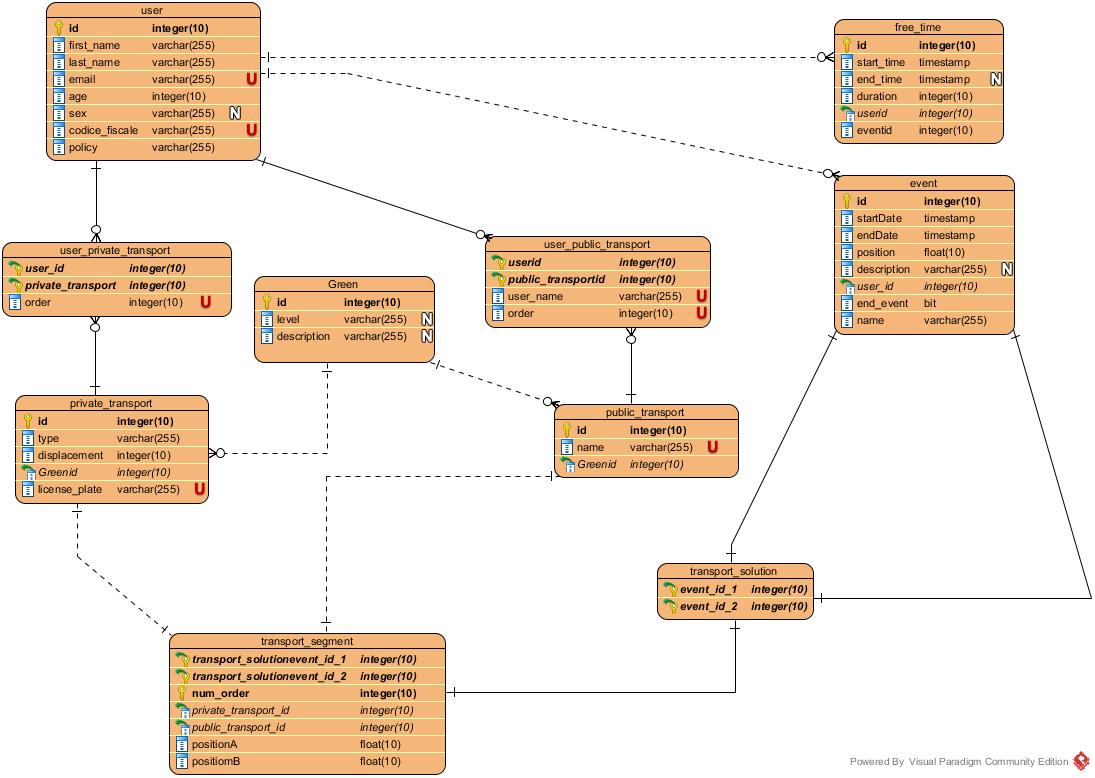
*Figure 2.4*

**Database**

The database tier runs in an external database service that allow us to store data more safely than in an internal db. We use InnoDB as the database engine: the DBMS has to support transactions and ensure ACID properties.

Access to the data must be granted only to authorized users possessing the

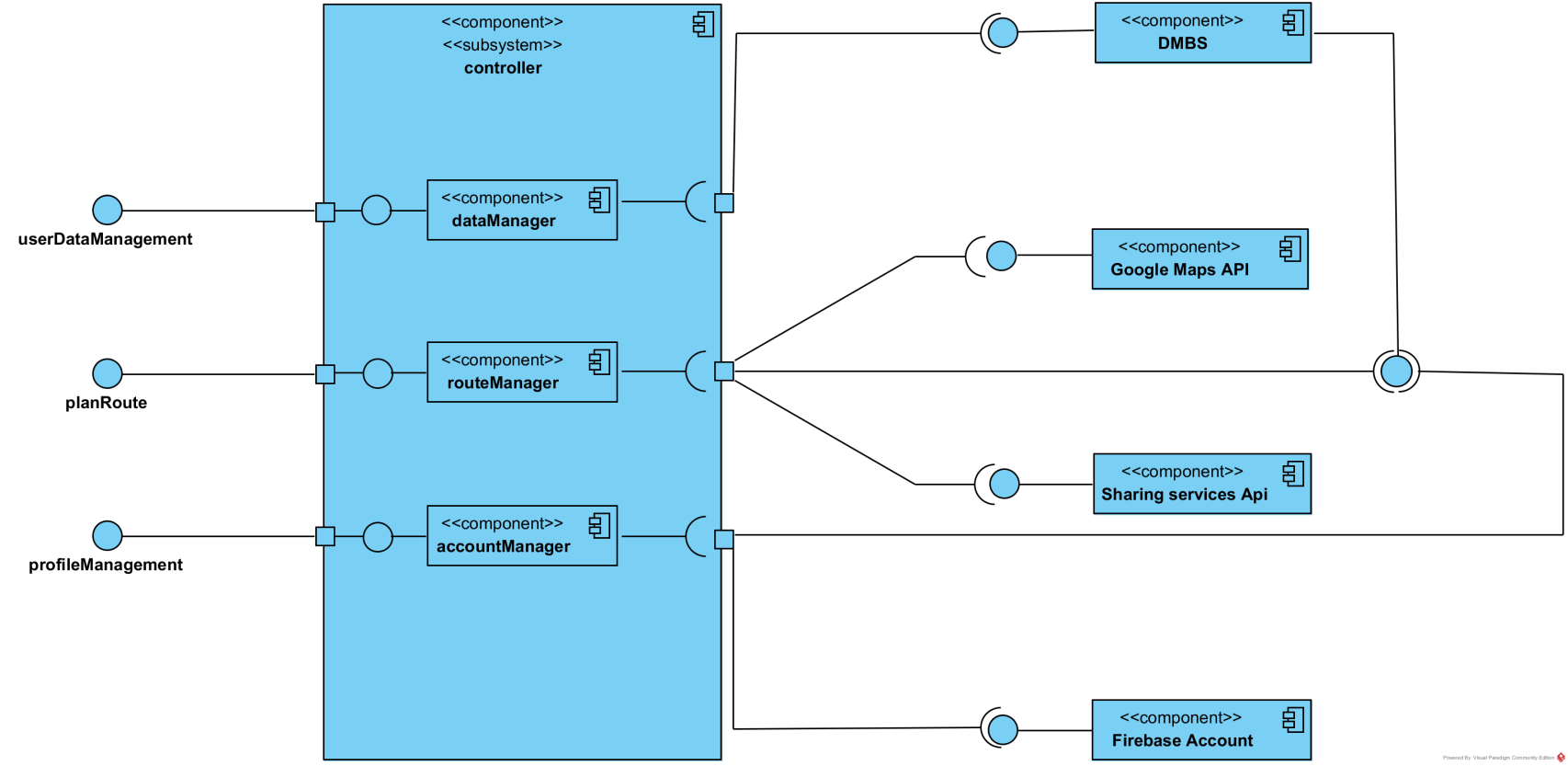
right credentials. Every software component that needs to access the DBMS.

****

*Figure 2.4: The Entity-Relationship diagram of the database schema.*

**Controller projection**

Controller subsystem is composed by three main components: *DataManager, accountManager* and *routemanager.* The first one concerns all the business logic about the management of the data that will be putted/getted into/from the database. The routeManager provides the logic about find the best solutions to link the events of the use with se Google Maps Api and Sharing services Api support . The accountManager has the goal to identify the user by Firebase Api.

******