Software Engineering 2 Mandatory Project Data4Help

Davide Damato, Luciano Franchin

Politecnico di Milano

January 16, 2019

Overview

- **The problem:** TrackMe is a company that wants to develop a software-based service allowing third parties to monitor the location and health status of individuals. This service is called Data4Help. TrackMe wants to use the data acquired through Data4Help to offer a personalized and non-intrusive SOS service to its users.

Goals

1. Collect user data.
2. Allow Third Parties to receive data collections.
3. Allow a user to monitor his own parameters.
4. Allow a user to receive first aid in emergency situations.

Domain Assumptions

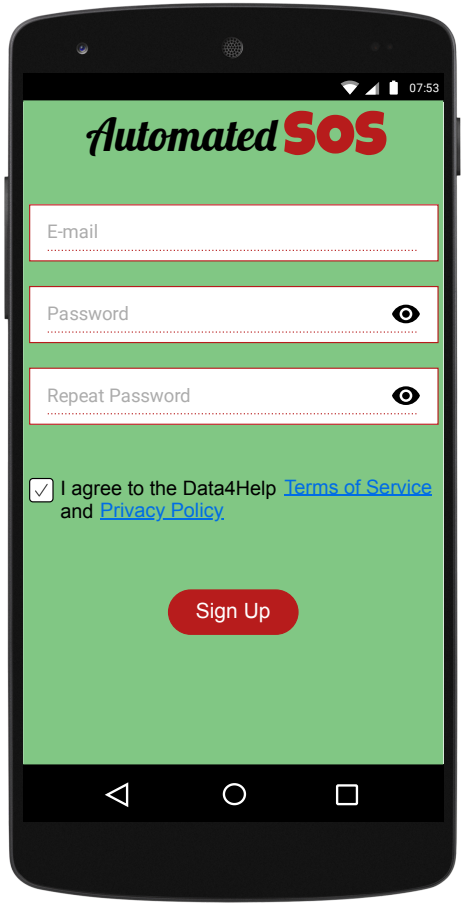
1. Once a detection device is paired with an user it will not be used by anyone else, if so, a disconnection procedure is necessary.
2. Every GsmDevice or GpsDevice can be uniquely identified with a code. (IMEI for example).
3. Detection device is attached to the user body when active so that measurements are valid.
4. Detection device is used properly by the user: regularly charged and worn as much as possible.
5. Any first aid service that proposes for AutomatedSOS service is sufficiently qualified and will give all the certifications or documentation needed.
6. Every qualified first aid service will have a communication channel to gather emergency information from AutomatedSOS.

Functional Requirements

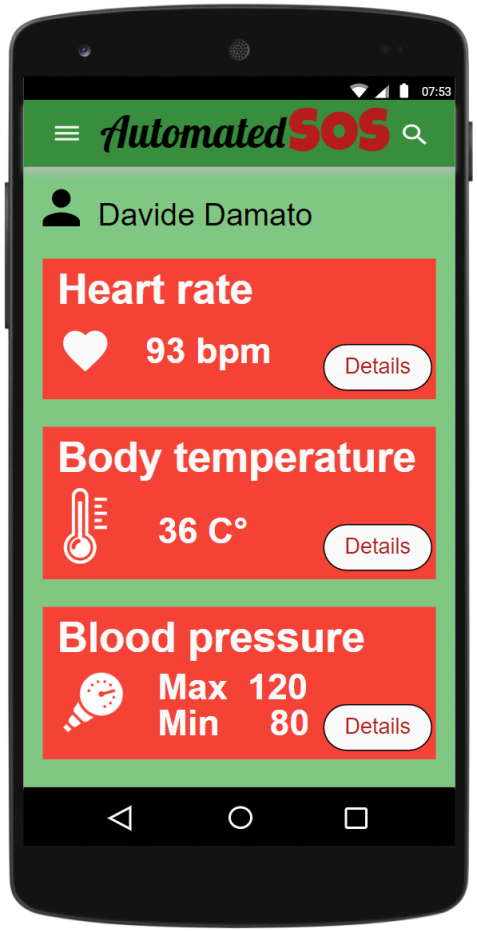
1. The system must be able to acquire user's health data.
2. The system must be able to save and store safely all user's data.
3. TPU must be able to select the desidered parameters to start a query.
4. User must be able to analyze and monitor their health parameters.
5. User must receive first aid in emergency situations when health parameters are lower than threshold values.

Mockups (I)

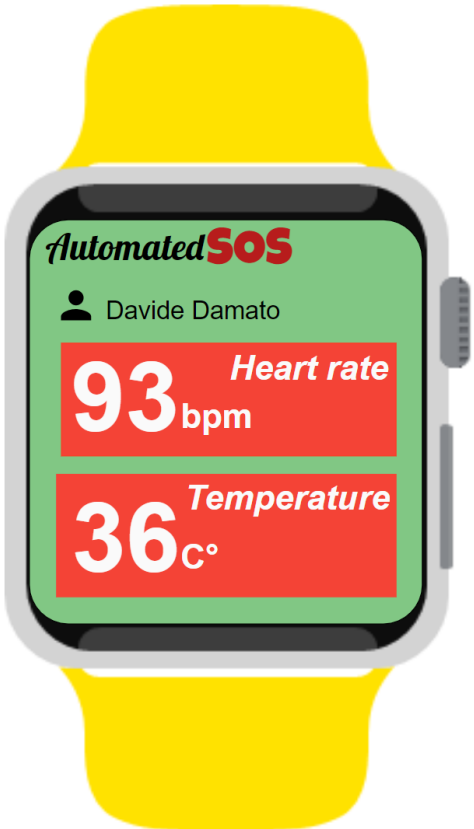
PU Registration



Smartphone Health Analysis

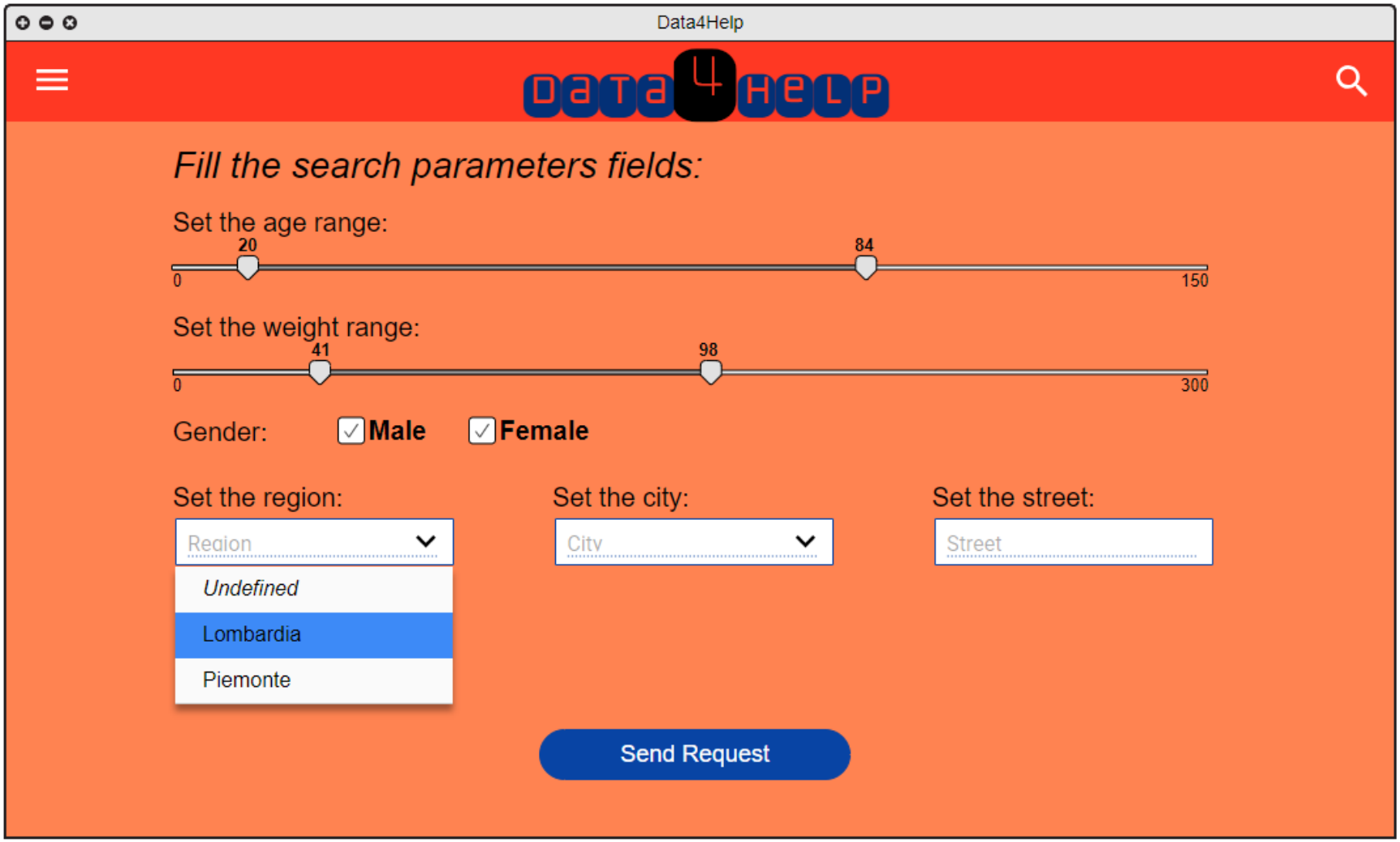


Smartwatch Health Analysis

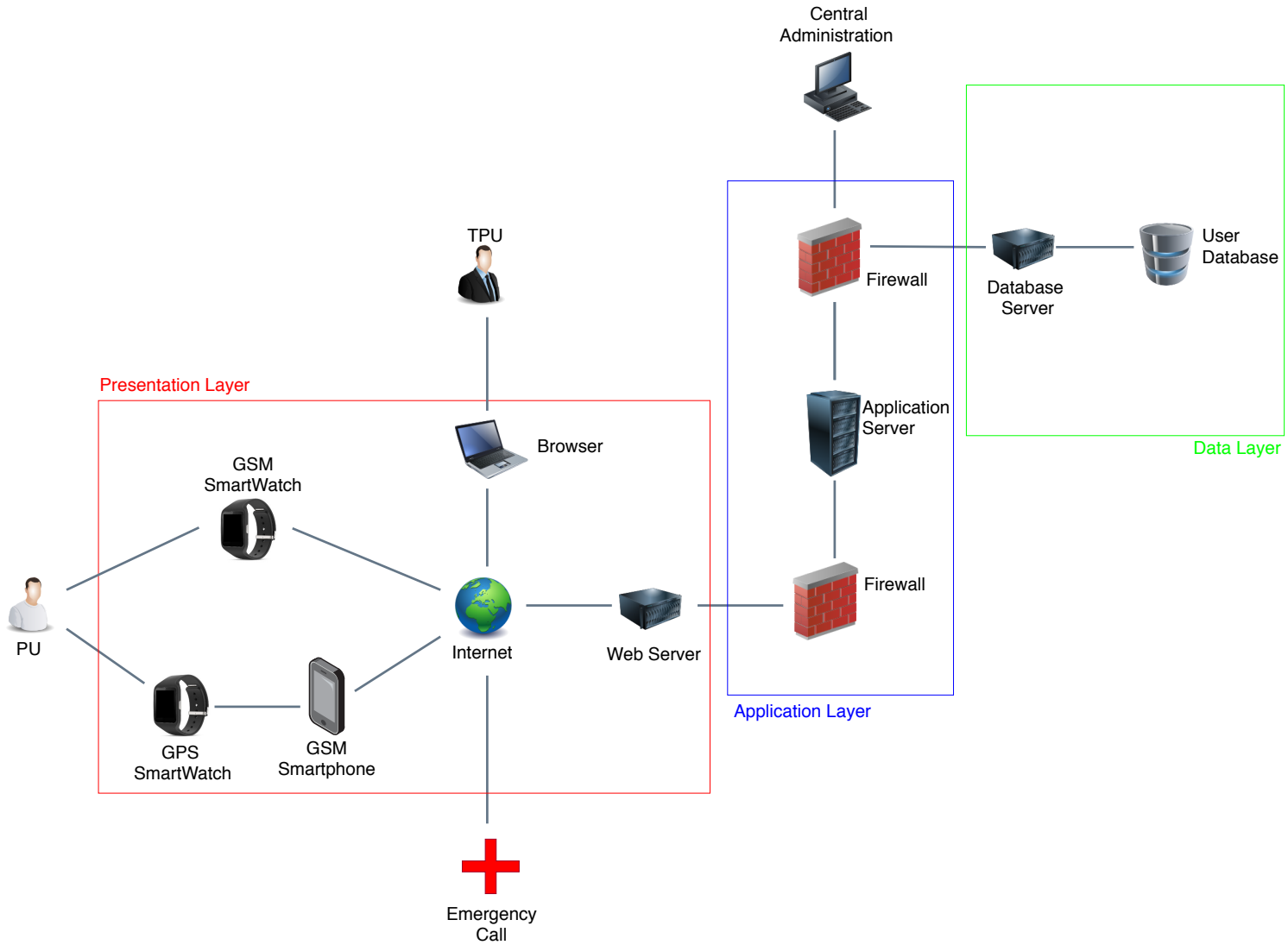


Mockups (II)

TPU Group Query



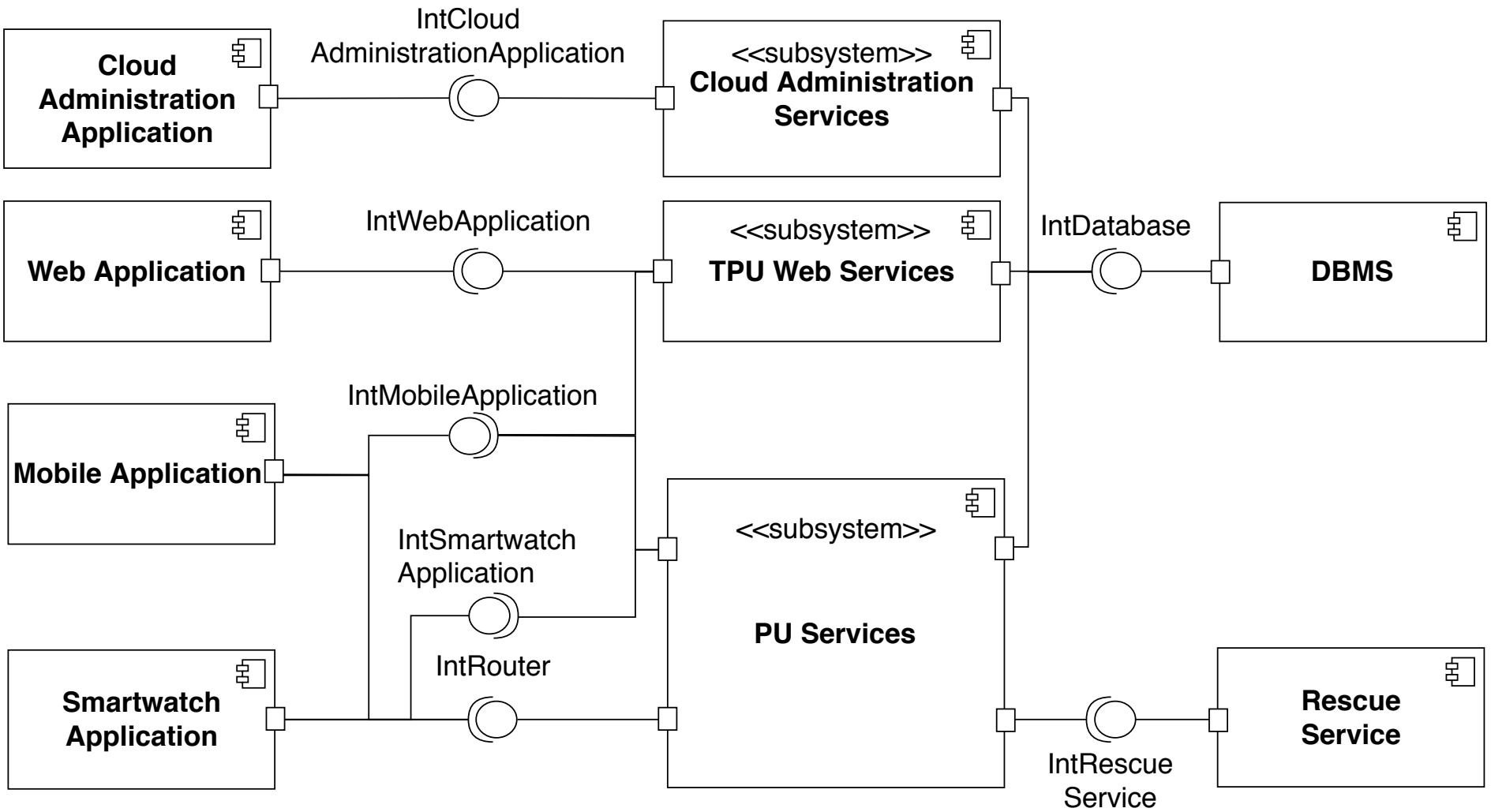
Architectural Design



Architecture: High Level Components

- **Web Application**: web page to allow TPU to perform queries.
- **TPU Web Services**: groups every TPU services functionality.
- **Mobile Application**: represents the software application installed on mobile OS.
- **Smartwatch Application**: represents the software application installed on smartwatch OS.
- **PU Services**: implements all the PU functionalities.
- **Cloud Administration Application**: externally developed application interface to manage main administration function.
- **Cloud Administration Services**: offers all the required services to control application and data layer.
- **DBMS**: manages every transaction between the components and the real Database where all the data is stored.
- **Rescue Service**:groups every outside first aid service that will be linked with the AutomatedSOS system.

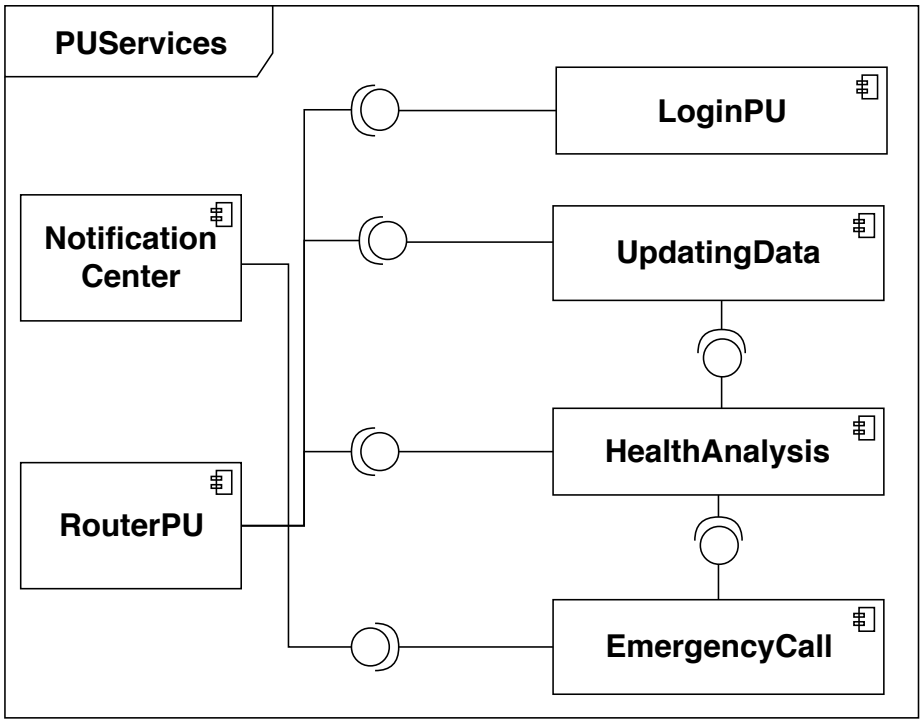
High Level Components

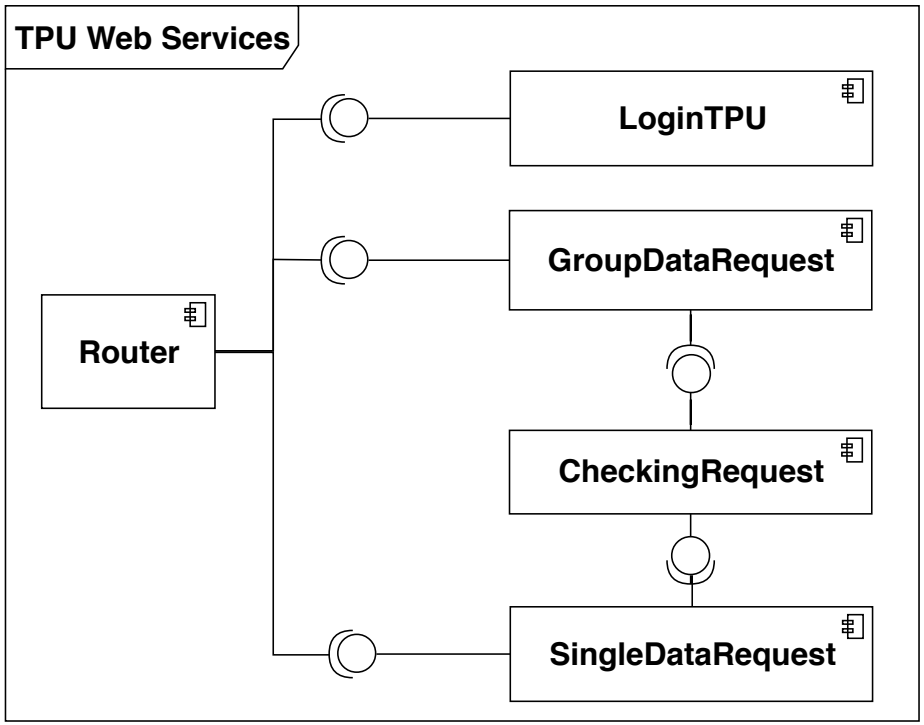


Architectural Styles

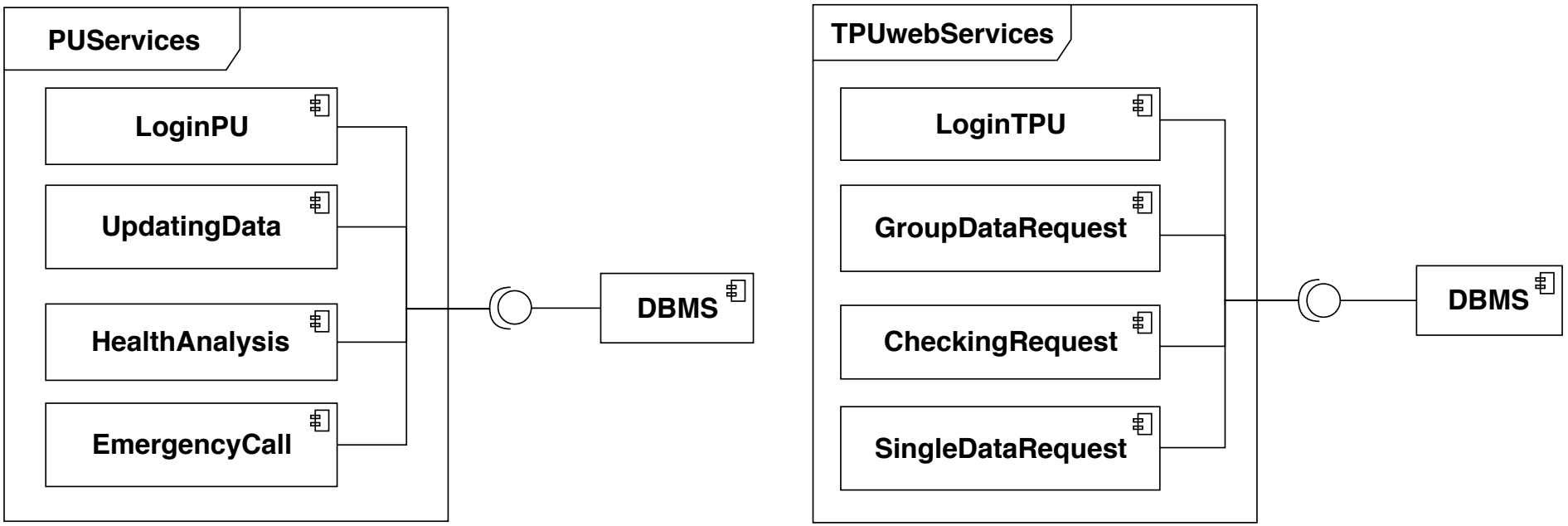
- **Layered Architecture**
  - Separation of concerns
  - Great clarity and flexibility
  - Different services run on different machines
- **Client/Server Architecture**
  - Business Logic implemented in servers
  - Clients used only for presentation purposes
- **3-tier Architecture**
  - Critical services are isolated and protected from external attacks

Integration of Components: Application Layer





Integration of Components: Data Layer



Integration of Components: Presentation Layer