UNIVERSITÀ DEGLI STUDI DI MODENA E REGGIO EMILIA
Dipartimento di Ingegneria "Enzo Ferrari"
Corso di Laurea in Ingegneria Informatica (D.M. 270/04)

Analisi, progetto e sviluppo di un'interfaccia di un API con Wearable Device in ambito e-health

Relatore:
Chiar.ma Prof. Sonia Bergamaschi

Correlatore:
Ing. Marco Pacchioni

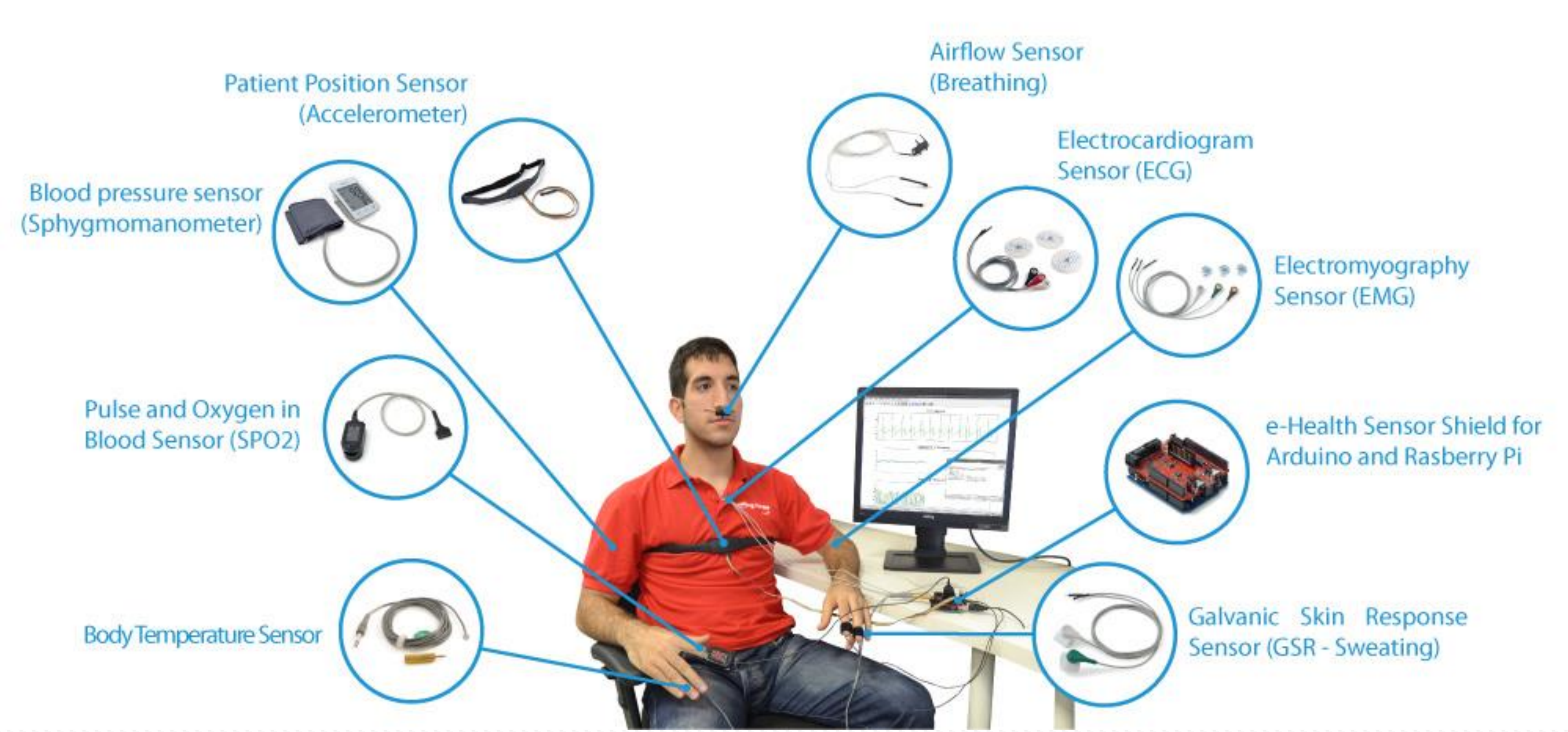
Candidato:
Matteo Gabrielli

Anno Accademico 2015 - 2016

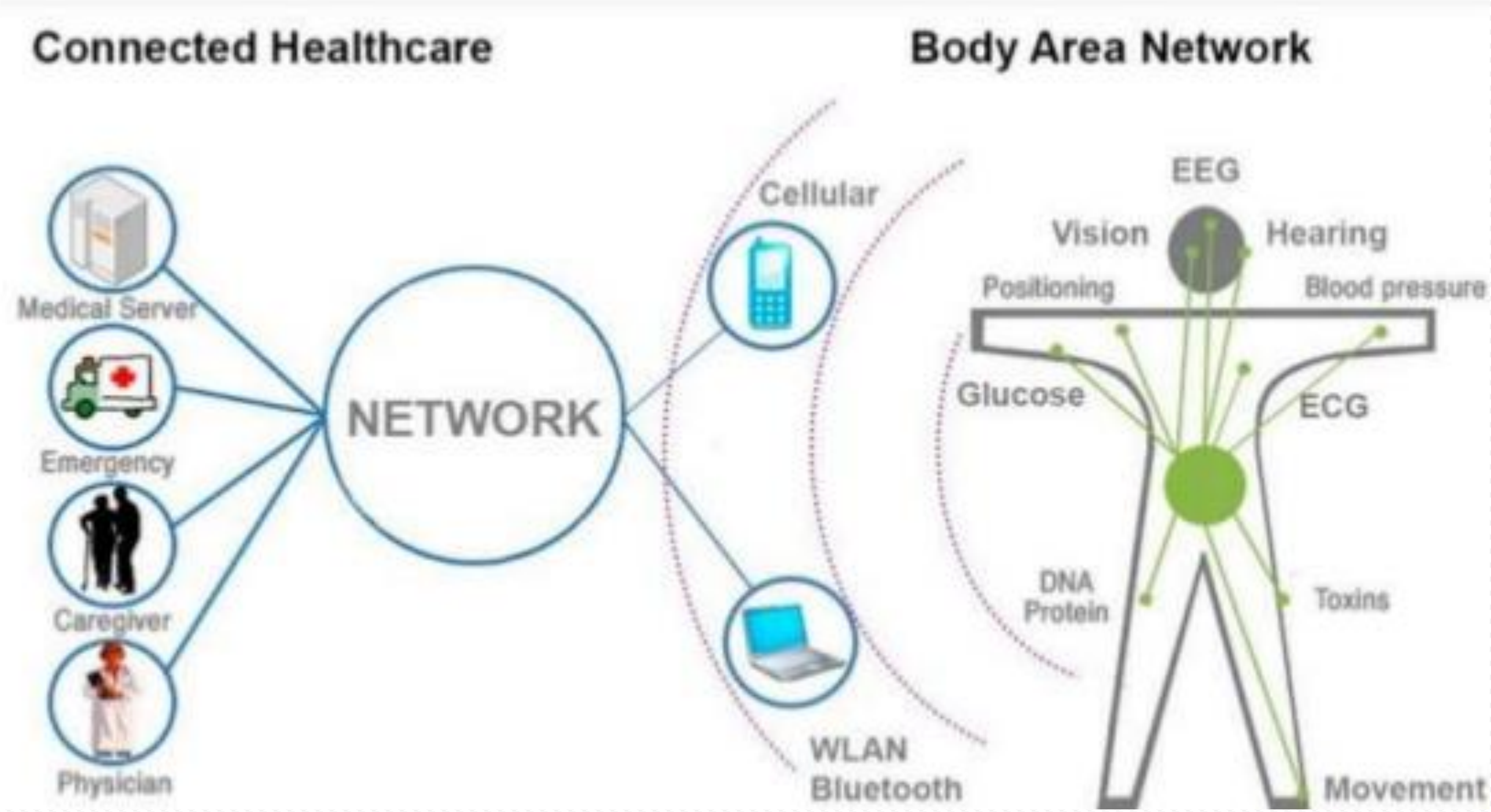
E-health

Per "eHealth" s'intende l'utilizzo di strumenti basati sulle tecnologie dell'informazione e della comunicazione per sostenere e promuovere la prevenzione, la diagnosi, il trattamento e il monitoraggio delle malattie e la gestione della salute e dello stile di vita.

Sensori e-health

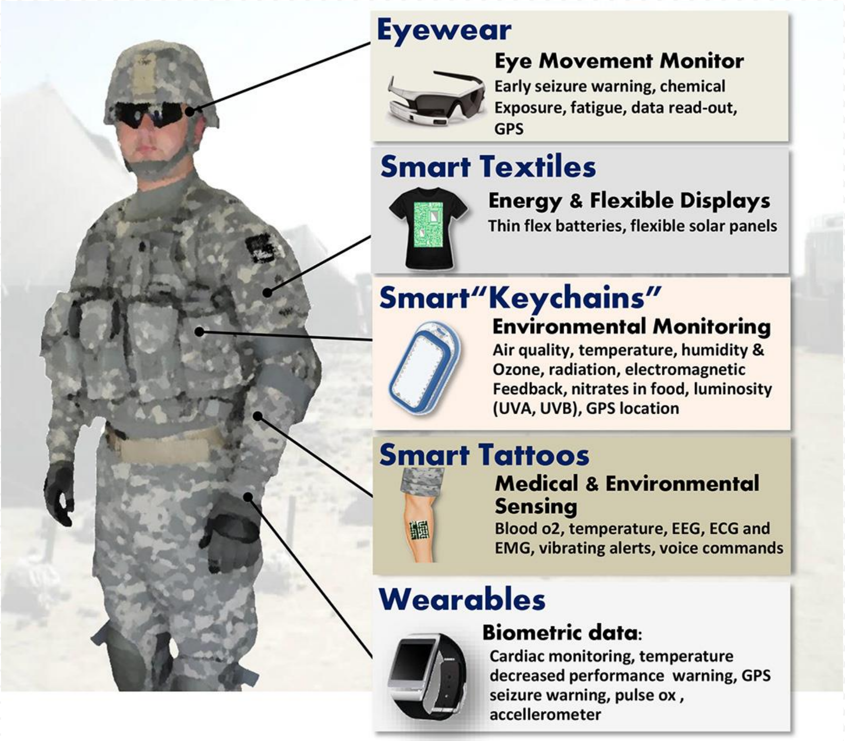


Connected human



Settori wearable technologies

- Salute
- Sport
- Personale
- Assicurazione
- Militare



Eyewear

Eye Movement Monitor

Early seizure warning, chemical Exposure, fatigue, data read-out, GPS

Smart Textiles

Energy & Flexible Displays

Thin flex batteries, flexible solar panels

Smart"Keychains"

Environmental Monitoring

Air quality, temperature, humidity & Ozone, radiation, electromagnetic Feedback, nitrates in food, luminosity (UVA, UVB), GPS location

Smart Tattoos

Medical & Environmental Sensing

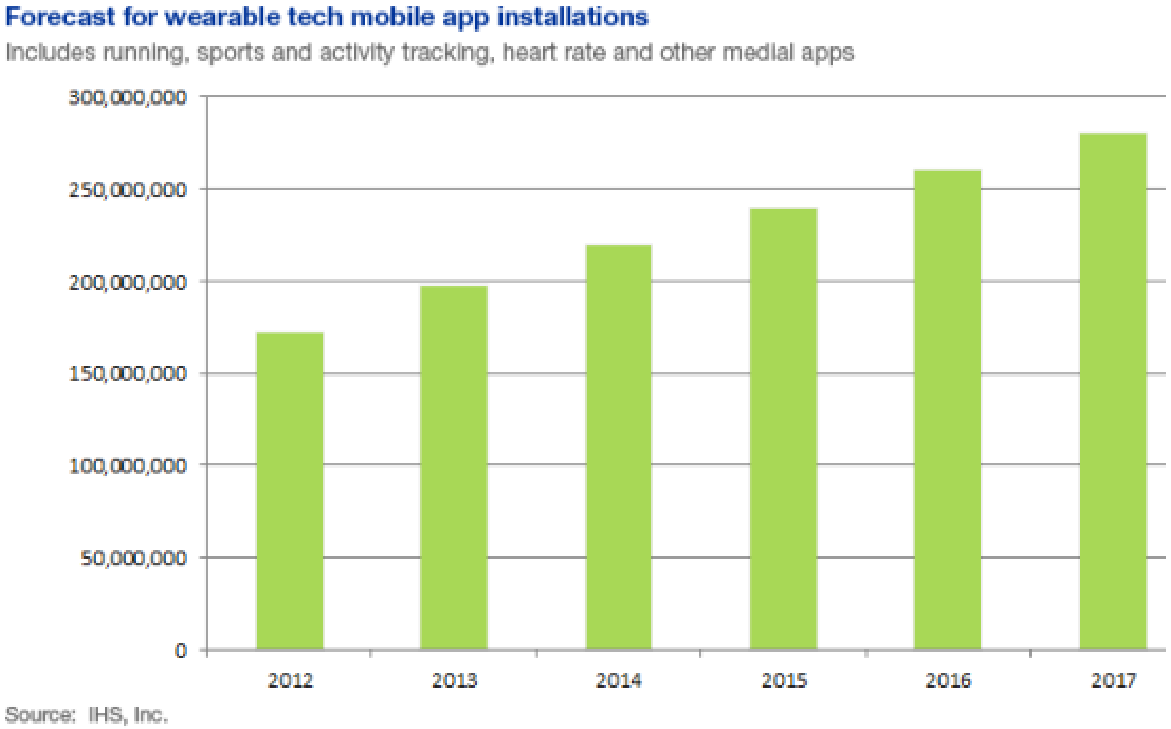
Blood o2, temperature, EEG, ECG and EMG, vibrating alerts, voice commands

Wearables

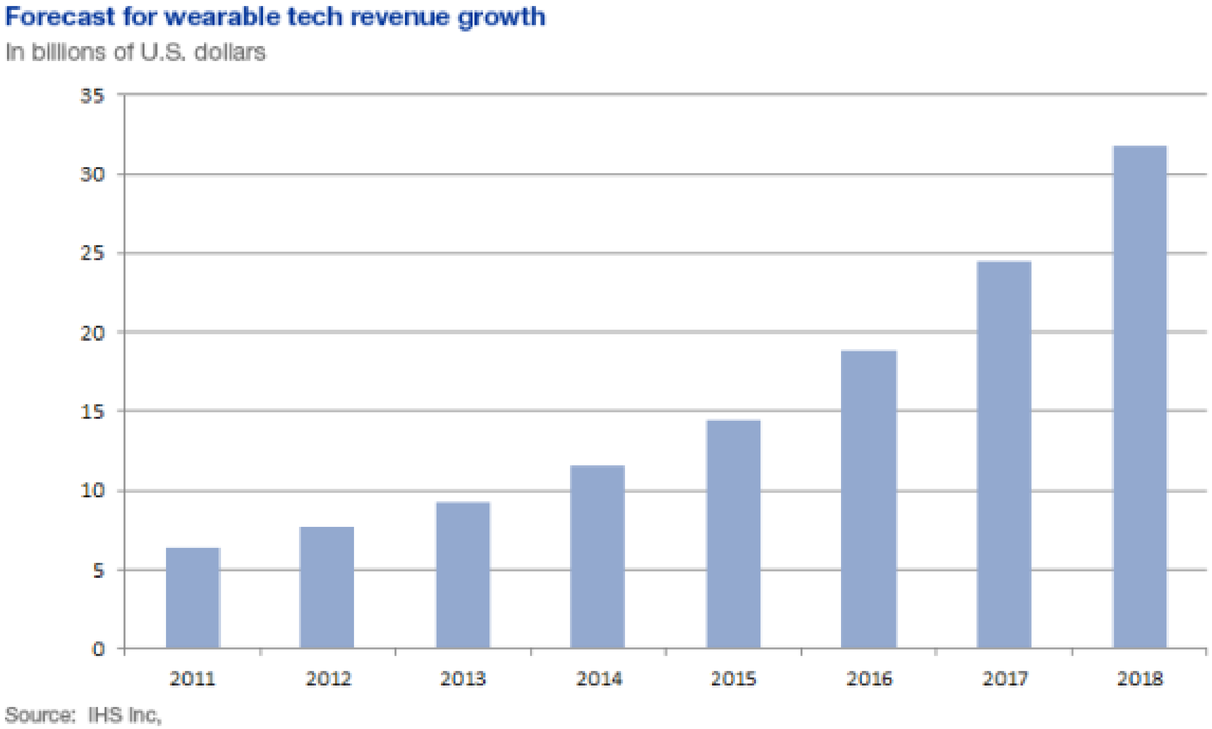
Biometric data:

Cardiac monitoring, temperature decreased performance warning, GPS seizure warning, pulse ox , accellerometer

Mercato wearable



Source: IHS, Inc.



Source: IHS Inc,

Progetto in ambito eHealth

DataRiver

open source data management

1. Analisi dei Wearable Device e studio API
2. Interfacciamento e reperimento dati
3. Elaborazione e visualizzazione

1. Analisi dispositivi e API

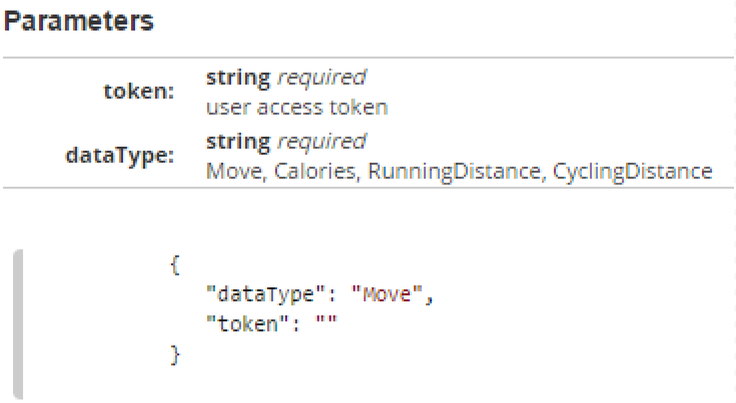
GARMIN™

TECHNOGYM®

Analisi Time Series

Retreive the user time and series data. Time and series data are available only for months for these categories: Calories, Move, Clycling distance, Running Distance and Time.
Response always contains all month days also if user has no results.

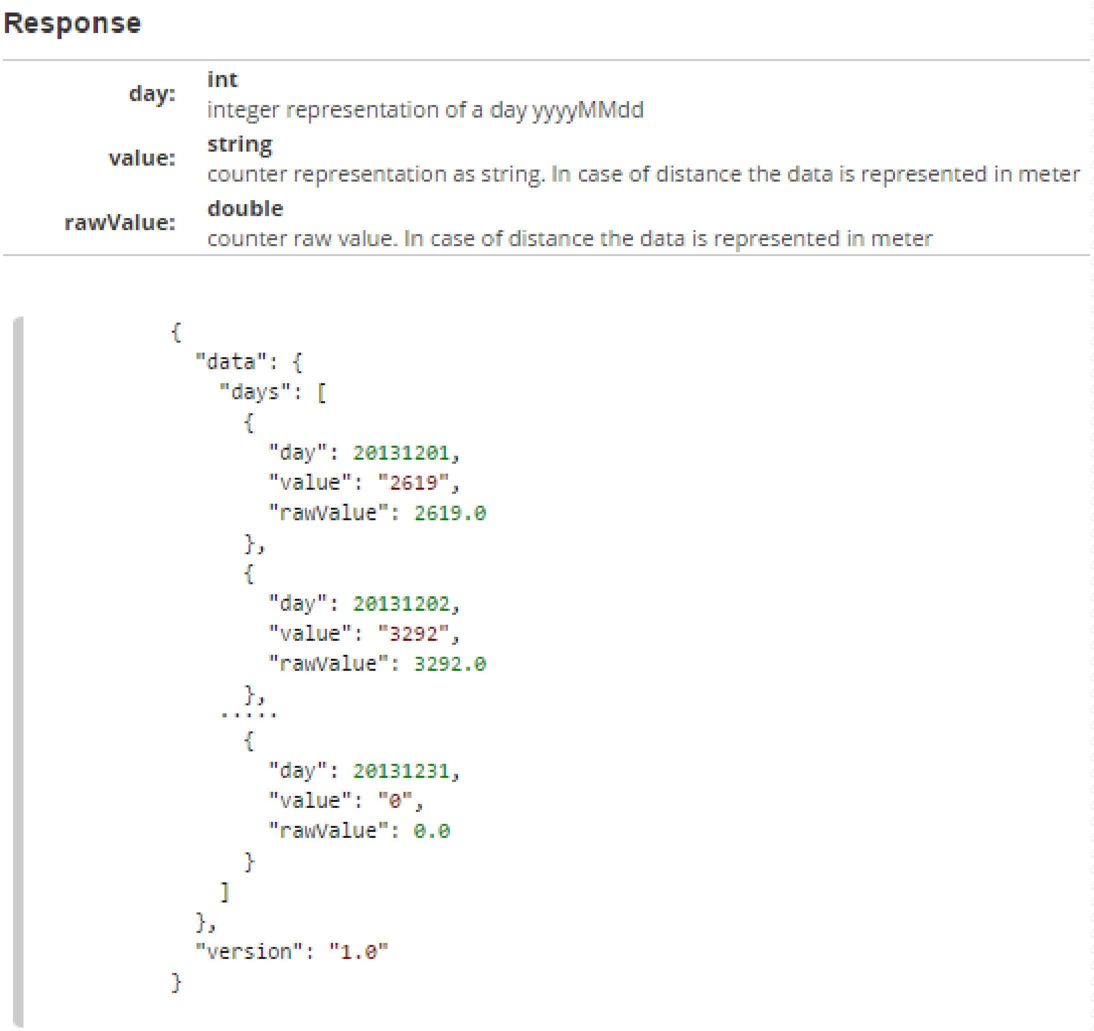
Url:
https://services.mywellness.com/api/[version]/ActivityStream/[TimeFrameId]/TimeSeries
HttpMethod: *GET*



Parameters

token:	**string** *required* user access token
dataType:	**string** *required* Move, Calories, RunningDistance, CyclingDistance

```
{
  "dataType": "Move",
  "token": ""
}
```



Response

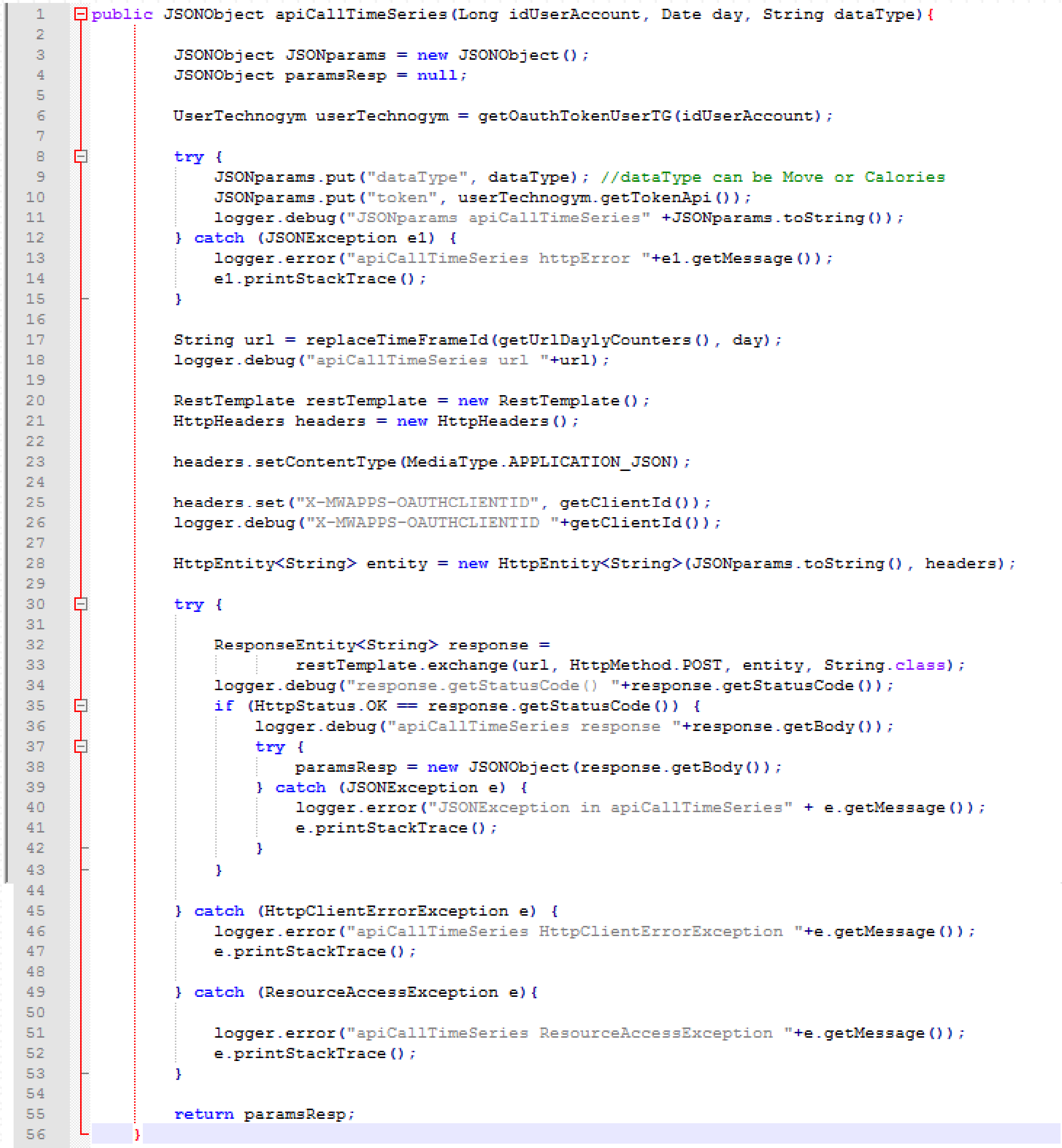
day:	**int** integer representation of a day yyyyMMdd
value:	**string** counter representation as string. In case of distance the data is represented in meter
rawValue:	**double** counter raw value. In case of distance the data is represented in meter

```
{
  "data": {
    "days": [
      {
        "day": 20131201,
        "value": "2619",
        "rawValue": 2619.0
      },
      {
        "day": 20131202,
        "value": "3292",
        "rawValue": 3292.0
      },
    .....
      {
        "day": 20131231,
        "value": "0",
        "rawValue": 0.0
      }
    ]
  },
  "version": "1.0"
}
```

2. Interfacciamento e pull dati

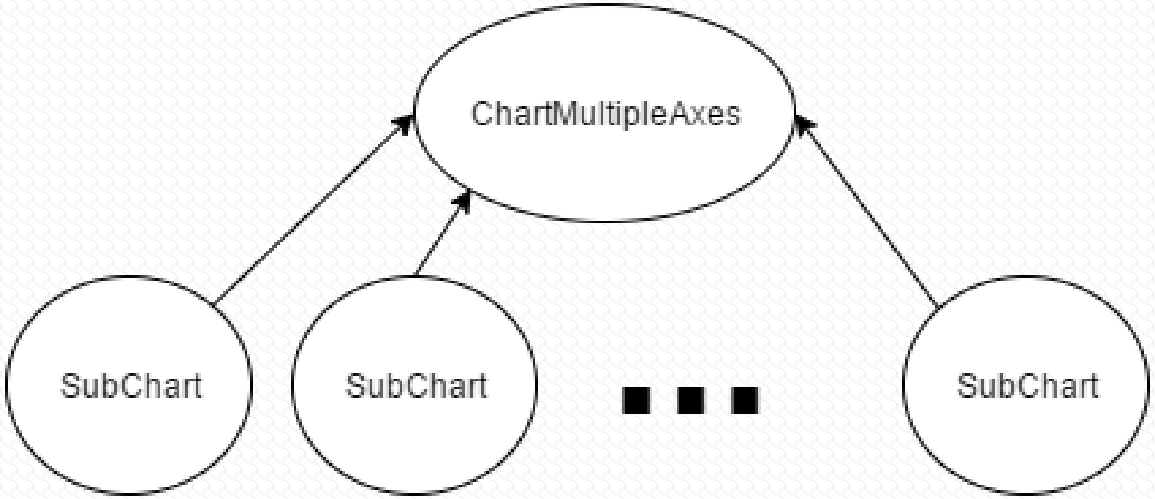
- OAuth 2.0
- Gestione JSON ritornato
- Richieste tramite API
- DAO (Data Access Object)

API call Time Series



```java
public JSONObject apiCallTimeSeries(Long idUserAccount, Date day, String dataType){

        JSONObject JSONparams = new JSONObject();
        JSONObject paramsResp = null;

        UserTechnogym userTechnogym = getOauthTokenUserTG(idUserAccount);

        try {
            JSONparams.put("dataType", dataType); //dataType can be Move or Calories
            JSONparams.put("token", userTechnogym.getTokenApi());
            logger.debug("JSONparams apiCallTimeSeries" +JSONparams.toString());
        } catch (JSONException e1) {
            logger.error("apiCallTimeSeries httpError "+e1.getMessage());
            e1.printStackTrace();
        }

        String url = replaceTimeFrameId(getUrlDaylyCounters(), day);
        logger.debug("apiCallTimeSeries url "+url);

        RestTemplate restTemplate = new RestTemplate();
        HttpHeaders headers = new HttpHeaders();

        headers.setContentType(MediaType.APPLICATION_JSON);

        headers.set("X-MWAPPS-OAUTHCLIENTID", getClientId());
        logger.debug("X-MWAPPS-OAUTHCLIENTID "+getClientId());

        HttpEntity<String> entity = new HttpEntity<String>(JSONparams.toString(), headers);

        try {

            ResponseEntity<String> response =
                    restTemplate.exchange(url, HttpMethod.POST, entity, String.class);
            logger.debug("response.getStatusCode() "+response.getStatusCode());
            if (HttpStatus.OK == response.getStatusCode()) {
                logger.debug("apiCallTimeSeries response "+response.getBody());
                try {
                    paramsResp = new JSONObject(response.getBody());
                } catch (JSONException e) {
                    logger.error("JSONException in apiCallTimeSeries" + e.getMessage());
                    e.printStackTrace();
                }
            }

        } catch (HttpClientErrorException e) {
            logger.error("apiCallTimeSeries HttpClientErrorException "+e.getMessage());
            e.printStackTrace();

        } catch (ResourceAccessException e){

            logger.error("apiCallTimeSeries ResourceAccessException "+e.getMessage());
            e.printStackTrace();
        }

        return paramsResp;
}
```

3. Elaborazione e visualizzazione



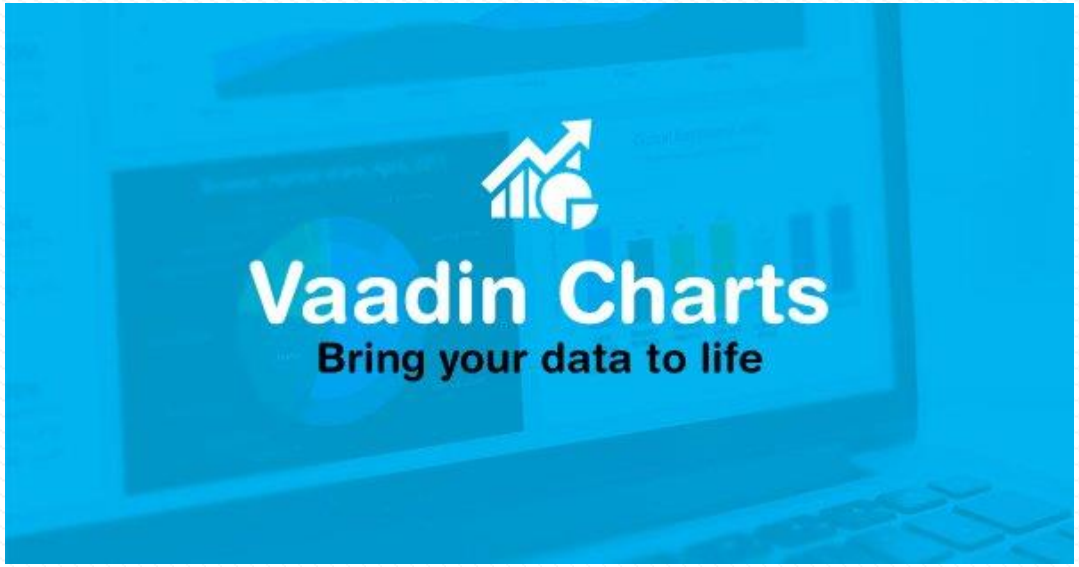
ChartMultipleAxes

SubChart

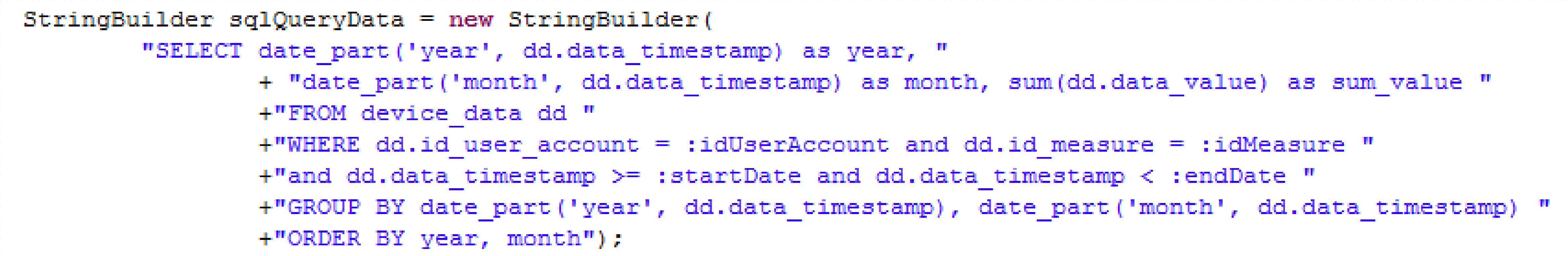
SubChart

...

SubChart

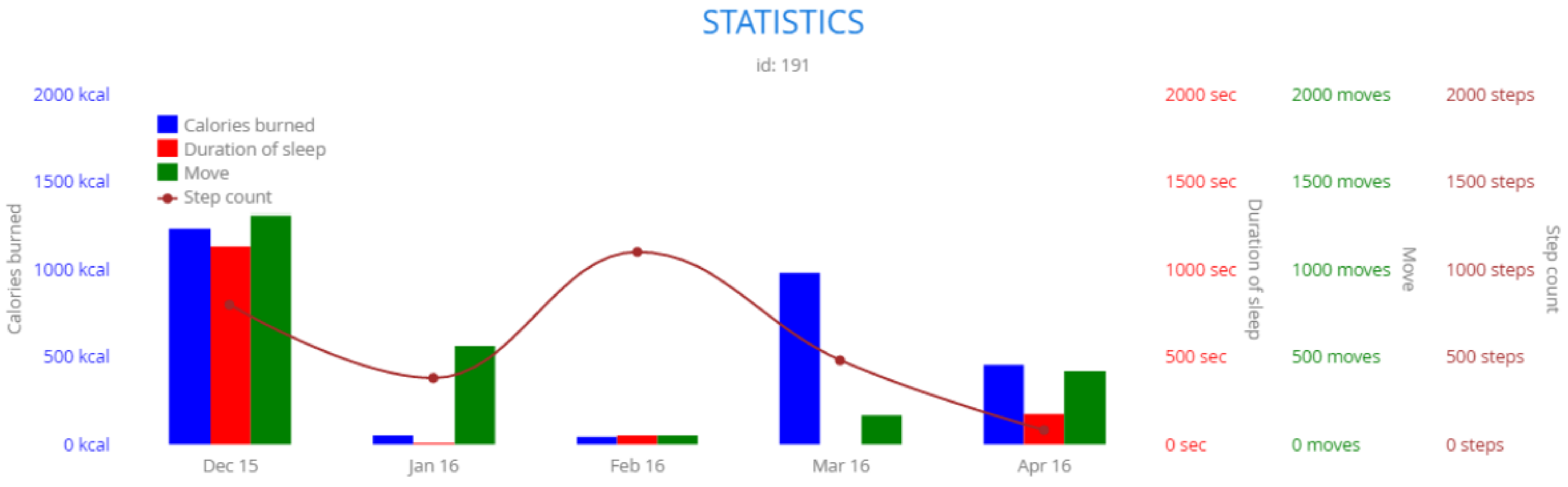


Device data SQL query



```
StringBuilder sqlQueryData = new StringBuilder(
        "SELECT date_part('year', dd.data_timestamp) as year, "
                + "date_part('month', dd.data_timestamp) as month, sum(dd.data_value) as sum_value "
                +"FROM device_data dd "
                +"WHERE dd.id_user_account = :idUserAccount and dd.id_measure = :idMeasure "
                +"and dd.data_timestamp >= :startDate and dd.data_timestamp < :endDate "
                +"GROUP BY date_part('year', dd.data_timestamp), date_part('month', dd.data_timestamp) "
                +"ORDER BY year, month");
```

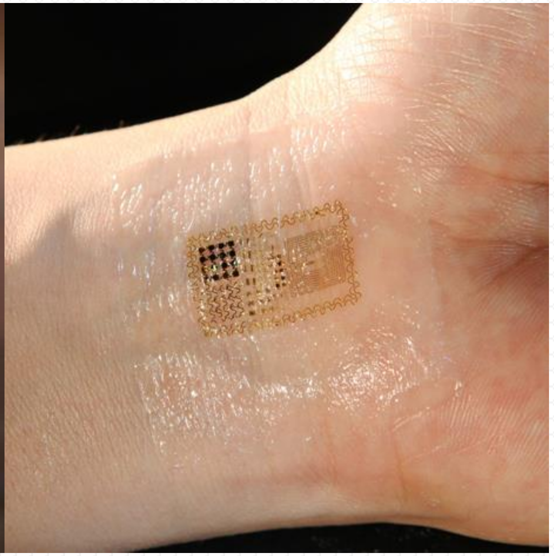
3. Elaborazione e visualizzazione



Conclusioni

- Risparmio costi per settore sanitario
- Sicurezza per l'utente
- Ubiquità
- Maggiori possibilità di sviluppo nell'IT

Sviluppi futuri



Grazie a tutti per l’attenzione

Per maggiori dettagli consultare la pagina web
http://www.dbgroup.unimo.it/site2012/index.php/published-thesis/triennale