

# Requirement Specification



## User Types

There are two types of end users and only the obvious maintainer roles (sales, devops and development).

- Runner – uses the app to make his sport time social.
- Analyst – crunches data gathered by the app.

## External Systems

We'll use following external systems:

Autoscaling cloud backend (AWS or GCloud) for APIs and app updates.  
Mobile application stores (App Store, Google Play Store) distributing the base product.

## Characteristics (Non-functional Requirements)

What qualities that the solution must have and in within which constraints must it operate to be effective?

- Safety – Dangers related to application usage must not be greater than dangers related to using Tinder or Pokemon Go.
- Legality – All data to be stored or sold must be anonymized. Solution must accord to information privacy laws. Therefore data is preferred to be stored in USA.
- Availability – The solution must be operational 24/7.
- Scalability – The base solution (Buddy for Runners) must be able to scale itself autonomously. The data vendoring system is considered secondary and can lag behind.
- Testability and maintainability – Code and infrastructure has to keep the highest standards.
- Approachability for end user – The application has to be intuitive and easy to get into.



# Capabilities (Functional Requirements)

The solution does:

1. Track **runners** and gather their characteristics.
2. Allow **runners** to draw routes and plan their **runs**.
3. Match **runners** with each other using location and their profiles
  1. Match strangers to **talks** where they can conversate about a new **run** suggested to them by Buddy.
  2. Allow **runner** to see his Facebook friends on the map and join them.
  3. Allow **runner** to see his friends public statistics.
4. Allow **runner** to mark his friends as observed to be notified about their improvements.
5. Vendor anonymized data to corporate customers.