# Introduction

UXprobe is a company with one important question. How do you users interact with applications? From answering this question the company builds the foundation of its business model, helping companys improve their applications. Uxprobe tracks how users move throughout and interact with an application as well as gathering user satisfaction data. Through this movement and satisfaction UXprobe hopes to understand how to better applications.

# Dataset

The origins of the dataset come from one customer of UXprobe. This customer is a transport company. Every truck has a tablet that runs a software developed in-house. It runs on an Android platform. This software is mainly a navigations system (TOM TOM), but also manages communication between drivers and the company, the assignments of their tasks and time management (cf importance of the regulatory context, mimimal resting time, etc)

Originally, the software was complex as final users (the drivers) were not involved in the development process. UXprobe system allowed to analyze the use and satisfaction of the drivers with the system in context and lead to a redesign of the software.

## Principles

When a user starts interacting with the software, he enters a *session*. A session can last for several hours, typically 7 or 8 hours (due to the limitations of the time a driver can drive a truck before resting). The dataset contains a field ‘sessionID’ that uniquely identifies a session. During that session, the user will want to achieve several actions, such as sending a message. This type of action is called a *task*. A task can be successful: the task is completed, or unsuccessful: the task is abandoned or an error occurred. The field ‘taskID’ contains a description of the task.

To perform a task, the user will interact with the system, and perform some operations, these are called *events* (the field is actually called ‘activityID’ in the dataset, and contains a short description of the event in plain text). Events have different types: ‘screen’, ‘feature’ (the feature is actually called ‘event’ in the dataset), ‘error’, or other. Type is a field in the dataset.

When an event is started, a timestamp is recorded, and is stored in the dataset under ‘starttime’. The granularity of the timestamps has changed in the course of the development of uxprobe logging system. In the beginning the granularity was coarse, so that events could be timed at the same moment, which can lead to some abnormal sequence of events when sorted. A field sequence contains a unique identifier for the record in the dataset, and can be discarded for the analysis. In general, a task is a form of container for several events, but some events can happen outside of the context of a task (example: an error can occur).

## Data quality

In some (tasks or events?), the client inserted “(driving)” in the description, if the event happened while driving the truck. This string should be removed, otherwise it would create two apparently different events, while only one is meant. For half of the dataset we received, the userID is missing. There seem to be indication that some (tasks or events?) are named differently, but refer to the same instance. Further evaluation of the data quality will be done while analyzing the data.

## Missing information

The hierarchy of the features of the system is not available, the logs contain events, screens and features. The structure of the software (such as the hierarchy of menus, screens, etc) is not documented. In general, uxprobe does not necessarily know what the actual feature is, but the customer knows.

The actual sequence of events is not directly available in the dataset, but can be inferred from the timestamp: an event is followed by another event if it has a timestamp that immediately follows

## Missing data

We have no information about the comprehensiveness of the data, and if they represent the most common use of the system, and if all records of the sessions received are included.

## Volume of data

Two files of approximately the same size were available, with a total of 1.2 million records and 160,9 MBytes.

## Description of the entities and their corresponding fields

### Session

The field name is sessionID and contains a unique identifier of session

Example of value: ‘55df6ce6-bac0-4bbb-b7ab-2e824d8a01a4’

### Record

The field sequenceID contains a unique identifier of the record, and can be ignored.

Example of value: ‘3343ae5e-997d-11e4-bfb2-f7c03b184ea7’

### Task

A field taskID does not contain an ID, but a short description of the task.

Example of value: ‘New Message Notification’

### Event

The field is called activityID but contains a short description of the event rather than an ID. In this dataset we identified XXXX different events values.

Example of value: ‘Notification: new message’

### Event type

The field ‘type’ contains the type of the event, which can take XXX values: ‘screen’, ‘event’ (where here it is ‘feature’ that is meant), ‘error’, ‘task’ (??? What is this one?)

### User

A field userID contains the identification of the user.

Example of value: ‘VEACESLAV OK-000000000046Y000-327700050402453’

# Goals

At the core of UXprobe is idea of helping project managers figure out how to make their applications better. Therefore this is the core of the data vizualization. How can we visualize the way people use an application in way that tells us how to improve it? Is their something hidden in the data that can only be discovered through visualization?

# Task

Uxprobes provide their customers with a dashboard that shows them the use patterns in their apps. Using standard pie and bar charts they can get an ideo of how people use their application.