Requirements Specification for

The Social Travel App

Project Acronym:	**CitySDK**
Grant Agreement number:	**297220**
Project Title:	**Smart City Service Development Kit and its application Pilots**

D4.1 Mobility Pilot Application and its SDK components – Pilot App SRS

Revision: draft 1

Authors:

Taco van Dijk (Waag Society)

Project co-funded by the European Commission within the ICT Policy Support Programme		
Dissemination Level		
P	Public	X
C	Confidential, only for members of the consortium and the Commission Services	

Change History

Date	Version	Description	Updated By
26 nov 2012	0,1	Initial Draft	Taco van Dijk
29 nov 2012	0,2	Changed architecture, to fit in cdk-endpoints	Taco van Dijk
3 dec 2012	0,21	Added privacy recommendations	Taco van Dijk

Document Approvals

Name	Role	Signature

Table of Contents

1 Introduction

1.1 Purpose

In this document we describe the software requirements for a yet unnamed mobile application, further referred to as the social travel app (TA). As part of this application we will develop a supporting mobility information aggregation infrastructure, further referred to as the intelligence service (IS).

TA is being developed as part of the City SDK project, work package 4; mobility. These requirements will serve as the starting point for development by the project team at Waag Society, and to communicate the requirements of the system to the replication partners in the City SDK consortium.

1.2 Document Conventions

In this document hand drawn wireframes are used to illustrate use cases in section 3. These wireframes are part of an interactive 'paper prototyping' application which can be found at the following url: http://popapp.in/projects/50aa3efd22078bdc1d000709/preview

1.2.1 Definitions

- Top down intelligence
  Mobility intelligence originating from (open) data sources and city services

- Bottom up intelligence
  Crowdsourced mobility intelligence

- Spatial context.
  Can be used to tag or filter the geographic relevance of a piece of intelligence.
  Examples of spatial contexts are:
  - specific geographic location to tag a piece of bottom-up intelligence
  - a geometry such as the province of Utrecht to tag a piece of top-down intelligence
  - a route between locations to filter messages for an enduser

1.3 Project Scope

TA is to be understood as a reference application that uses and validates the appropriateness of the interfaces, components and specifications that are defined for the mobility domain in work package 2 of the CitySDK project. The aim is to replicate TA in the cities of Helsinki, Rome and Istanbul.

1.4 References

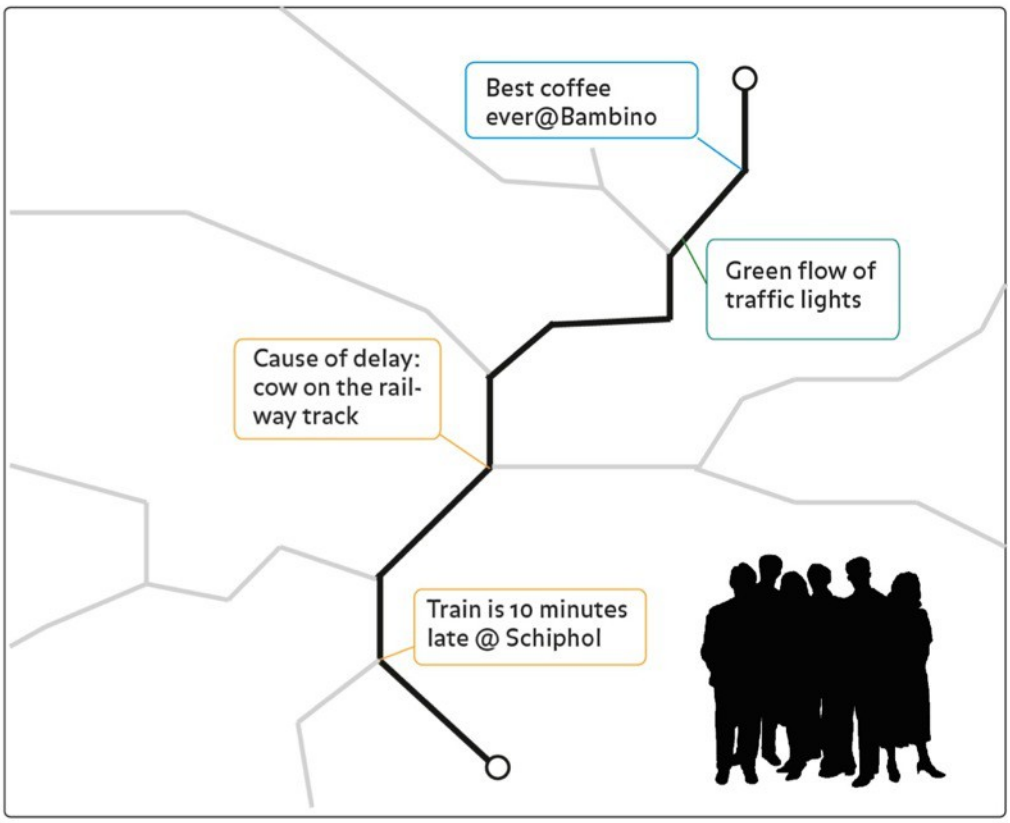
For more information on the CitySDK project, and how TA fits in see: http://www.citysdk.eu

TODO: link to document WP2

TODO: link to document gebruiksonderzoek

2 System Description

Based on the insights of the qualitative research on user's needs within the domain mobility Waag Society has developed the concept of an 'Intelligent Social Traveling'- system. This system should support the traveler during his journey and is based on the idea of people serving as information agents. By sharing their travel experiences travelers may help others, structure the traveling chaos, connect with fellow travelers and enhance public services.



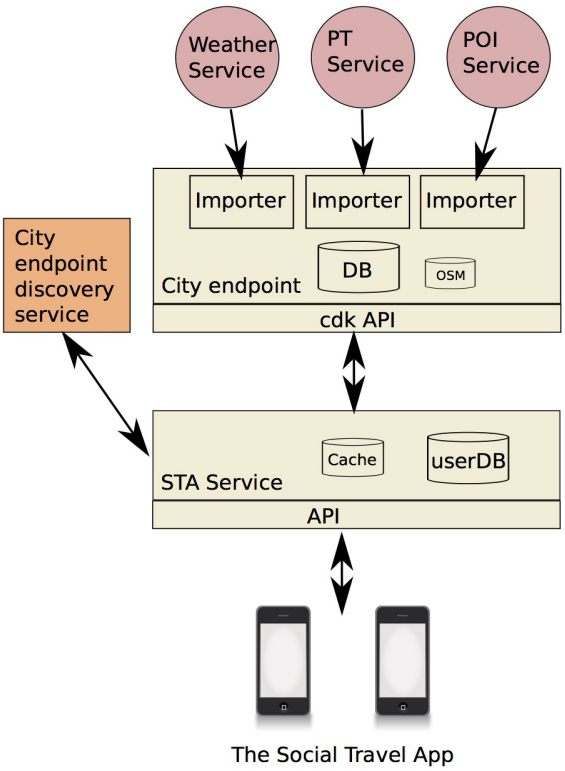
2.1 Problem description

People traveling through the chaos of the city are mostly self-dependent.
We want to use technology to stimulate communication, cooperation, reciprocity and social cohesion between citizens in order to make the experience of navigating through the city a more pleasant and useful activity.

2.2 Our solution

Instead of focusing only on the artificial intelligence possibilities of mobile computers such as route planning algorithms and turn-by-turn navigation,
we want to leverage a relatively untapped source in the mobility domain: human intelligence.

The following diagram gives a high level overview of the system.



The Social Travel app and the STA service are within the scope of this document. The City endpoint and discovery service are also developed in the CitySDK mobility work package, but should be regarded as external from this document's point of view.

2.2.1 The social travel App & Service

The service enables people to share intelligence about daily mobility issues in the city. A central notion is the route - defined as a series of locations between point A (start) and B (end). Each person that travels at a given moment has a unique route, but parts of the route overlap between persons and / or time.

The STA service enables all moving citizens to contribute information related to a position in their current route to a central stream of intelligence, which amounts to a view on a database of intelligence items indexed spatially and temporally.

This 'stream' may be filtered by using the route of another person as input. The result of this filter is a collection of intelligence that is current in the sense of time and place.

The information that we are targeting may be anything related to mobility.
For example information about: a bridge being open, a road being closed, a new coffee place that has been opened, a beautiful rainbow that is visible at a certain location, a great discount at the truck stop, traffic flow data, available parking space etc.

In addition to the bottom up intelligence provided by crowd-sourcing, we want to provide top-down intelligence originating from public services/institutions relevant to mobility that are available in the specific city where The App is used.

For example intelligence from a public transportation company via the city endpoint can be pushed/pulled in to the stream, as long as the relevant geographic area and time are provided. Same goes for weather information etc.

In order to promote and stimulate engagement, reciprocity and social cohesion we want to introduce some form of gamification (for example to accumulate likes). A traveller may receive points/karma for each piece of intelligence that is shared, even more for intelligence that is liked/thanked.

3 Functional Requirements

3.1 STA service

3.1.1 API

3.1.1.1 Post intelligence item

- required meta-data spatial index: location / region, index timestamp
- optional meta-data such as source, category, team, privacy level (partly application defined)
- optionally transform by security / group membership fields header (anonimize / aggregate / combine / encrypt)

3.1.1.2 Filter / retrieve intelligence items

- by timespan / route / geometry
- optionally by other metadata (category team)
- by security level / group membership

3.1.1.3 Link intelligence items
For example link public transport message (“5 minutes delay“) to crowd-sourced message (“10 minutes and I’m still waiting”)

3.1.2 City endpoint client

This module is responsible for pulling data from the endpoints to the cache.
And for pushing data from the cache to City endpoints.

For example pulling dat originating from a weather service or public transportation service,
or pushing comment data from crowd-sourced intelligence items as comments.

3.1.2.2 Query & filter data sources through City endpoints

3.1.2.3 (Optionally) Transform data to intelligence by adding spatial and time context metadata, before they are added to the cache

3.1.2.4 Push comments to the endpoints

3.1.3 Cache (CSI)

The cache contains all intelligence items that are currently relevant to the travelers of the city.

Items are relevant when they are spatially relevant for the routes of one or more travelers, and are within the time constraints. The default behavior will be that new intelligence expires after 12 hours before it is removed from the cache. But when a traveller 'likes' the item, the timer is reset to the time of the like and thus the item will remain in the cache longer.

3.1.4 User database

This database is responsible for persisting membership, trajectories, karma etc.

3.2 Social travel application (TA)

3.2.1 Select / define / record / import route

- import from route-planner
- record by enabling GPS and traveling from start to end

3.2.2 Pull intelligence from CSI

- providing current route and timespan as argument

3.2.3 Client-side intelligence filter

- filter on categories, modes of transportation, tags, other metadata without doing an extra request

3.2.4 Social actions on intelligence

- thank / give karma
- export to external social network like Twitter or Facebook.

3.2.5 Send intelligence to the CSI

- while providing current location and timestamp
- and providing extra metadata such as category, team etc.

3.3 Use Cases

The following is a description of the use cases for the social travel app. In the first version of our system there is only one role, the role of the traveler, the mobile enduser.

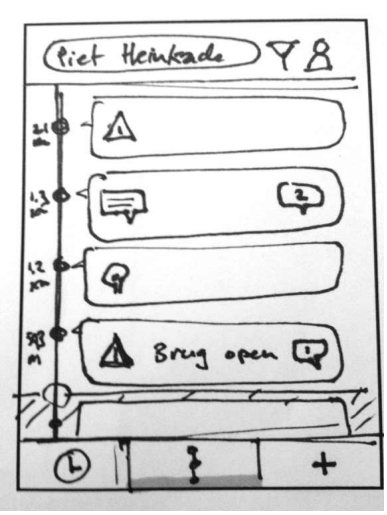
3.3.1 UC 1. Select a destination.

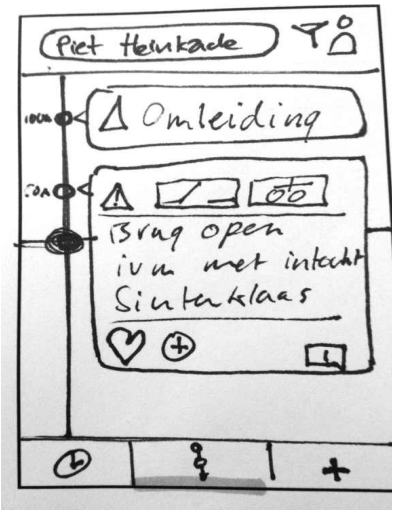
To keep things simple and fast, and to discriminate the functionality from a route-planner, the only thing the user has to specify is their current destination. This can be selected from a list with previous destinations, or typed in with the on-screen keyboard.

A trajectory consisting of relevant locations paths and symbols, is calculated/approximated by the app based on the current location of the user, and earlier trajectories of the user/and or shared trajectories uploaded by other users of the app. The first time this trajectory will be coarse, rectangle containing start and end position. The app will record the path the person is traversing to improve the granularity of the destination in the future.

3.3.2 UC 2. Read the list of relevant social intelligence.

The stream is displayed as a list of intelligence items. This list may be filtered and sorted.





For each piece of intelligence the following information is displayed:

- Main category: Disruption or Point of interest.
    - The disruption category encapsulates all information about things that may have a negative impact on your journey. Bad weather, congestion, delays, open bridges, etc.
    - The POI category contains information about things that might have a positive impact on your journey. Beautiful views, pieces of art, discounts, a new interesting place, a popular event, an available bus seat, free parking spaces, a free wifi access point etc.

- Origin category: (verified) Service Provider or Travelers
    - The Service Provider category indicates that information comes from an 'official' data provider such as public transport companies or weather stations.
    - The Traveller category indicates that information was crowd-sourced from travelers that share a parts of their trajectories with you.

- Tags (in the form of icons for the most common). These may include tags that specify the subtype of disruption, subtype of POI, or relevant mode(s) of transportation.

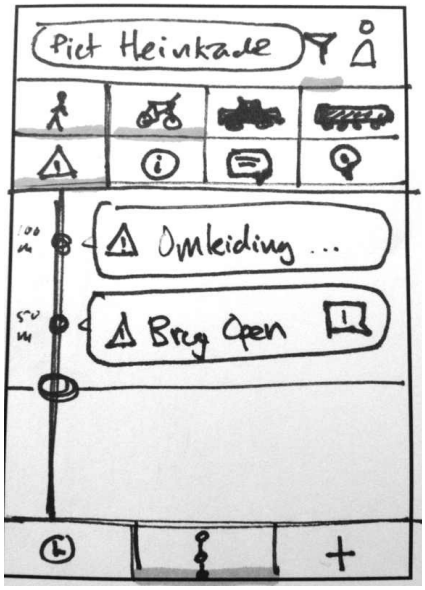
- Body: This might be in a structured form (text with pictures / html) when originating from a verified data provider. Or a tweet like free form text message when originating from another traveler.

The user has two sorted views on the stream of information. The views may be switched by

selecting one of the two buttons in the toolbar at the bottom of the screen.

- The 'time' view sorts all relevant messages so that the most current messages are displayed on top of the queue.

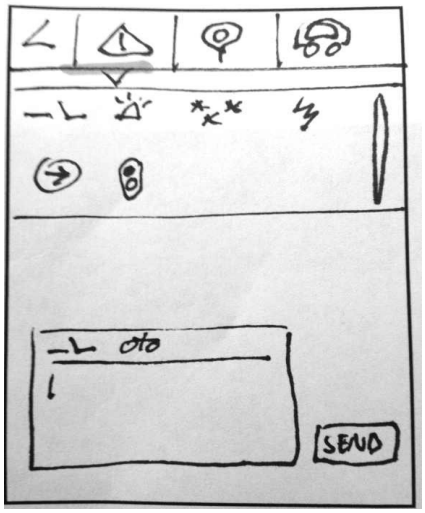
- The 'route' view sorts all relevant messages by distance to the destination. (information that is geographically close to the destination will be shown on top of the queue). Next to each message the distance to the current position is plotted for reference.



The user has the option to filter the list of messages by toggling the buttons for information type (disruption/poi), origin and transportation modes. (walk, cycle, car, public transportation).
This filter is persisted for each individual destination.

3.3.3 UC 3. add a social intelligence item.

The user has the ability to add a new intelligence item for the current time and place, based on the system clock and current GPS coordinate. The view to add a new intelligence item is activated by touching the '+' button in the lower toolbar



In this view, the user has to select the category, tags, and optionally the relevant modes of transportation in the form of icons as meta-data to the message.

Use of hashtags in the message body is allowed as an alternative method for free-form meta-data.

3.3.4 UC 4. social features: like, comment and share.

These functions are all initiated by touching the buttons in the footer of an intelligence item, and work as would be expected from other social applications such as facebook or twitter.

3.3.5 UC 5. Register / log-in / profile management.

To be further specified during development.

3.4 Data Dictionary

The following entities can be identified specifically for the app: destination, trajectory, and intelligence item (which is a container format for a collection of mobility specific information types).

3.4.1 Intelligence item

Todo: specify further

Attribute	Type	Optional?	Notes
<Attribute Name>	*<Data type of the attribute>*	*<Y or N>*	*<Explain any specific restrictions or rules applcable on this attribute>*

3.4.2 Trajectory

Todo: specify further

Attribute	Type	Optional?	Notes
<Attribute Name>	*<Data type of the attribute>*	*<Y or N>*	*<Explain any specific restrictions or rules applcable on this attribute>*

4 External Interface Requirements

The harvester component has to interface with the endpoint discovery service, to be specified as part of CitySDK wp2.

The harvester component has to interface with the CitySDK mobility API v0.2 as described by Tom in his document.

This API will be used to consume (open) datasources and services in the following domains:

- mobility services (public transport etc.)
- weather (percent rain, temp, wind)
- poi (points of interest)

4.1.1 External data sources

Within the scope of the pilot the intention is to initially integrate the following sources:

- gtfs from ov9292
- gvb realtime information
- ns api (optionally)
- openov / govi (optionally)
- poi from osm

- poi from arts holland (optionally)
- weather information from buienradar.nl & knmi (optionally)

5 Technical Requirements (Non functional)

5.1 Scalability

With regards to the intended number of users in the Amsterdam pilot, and the projected load scenarios, the intention is for the system to be able to serve 20.000 queries / day (in large part during the 4 peak traffic hours).

5.2 Security and privacy

The mobility domain has a privacy sensitive nature, specifically with regards to the location tracking of travelers.

In order to create a viable offering for the user we will have to build a simple, transparent system that can be understood and trusted by the people that are using it.

In order to build trust with the users of our system, the system can make use of the following strategies:

- Anonimization & aggregation, so that route information may be shared safely without disclosing personal information.
- Encryption, for all data that is privacy sensitive, but must be persisted on the server in order for basic functionality
- Open source / disclose security policies & practices
- Permit the use of unverified (anonymous) avatars / aliases.
- Give control to end-users over private data (at least a delete private repository option)

5.3 Multi lingual Support

The system should be language agnostic, since portability is a prime goal of the overall project. English, turkish, Finnish, and Italian should be at least be supported because these are the replication partners.

6 Open Issues

<There could be open issues even at the end of the requirements elicitation. List of all of them here so it can be tracked and closed later. Some of these issues may later become project risks as well.>