BusTracker Requirements

Ewan Tempero

15 September 2011

Overview

BusTracker is a system that has been requested for the Auckland region by the Auckland Regional Council.[1] The initial brief for this system is given below.

> *ARC wants a system called BusTracker that tracks buses. It wants to add equipment such as GPS to all of its buses so that it can track where they are to within 20 metres. They will use this information to provide estimated arrival times of buses at bus stops.*
>
> *The unit to be placed on each bus consists of a Global Positioning System (GPS) receiver, a radio transmitter, and other bits of hardware and software. It can transmit the location and speed of the bus, along with the bus' identifier to the BusTracker system on a regular basis.*
>
> *The planned displays will have a radio or GPRS receiver and room for display four or five bus numbers and times at the same time. The displays should repeatedly scroll through all the buses whose estimated or scheduled arrival times at that stop are sometime in the next hour.*
>
> *The displays that show the arrival times will initially installed at major bus stops, where a number of bus routes converge. There are typically 20 or more buses that stop at these stops during peak travel times. Once the system has been completed for the major bus stops, displays will be progressively added to other bus stops as the budget allows.*
>
> *The bus company also would like to eventually allow bus patrons to get estimated arrival times for all buses at all times via their web site, and also via phone. The phone access could be "push", via SMS, or "pull" via an app installed on the phone.*

System Architecture

Some aspects of the system are constrained as follows:

- There is a single radio receiver for the bus transmissions for the city. It is located in place that covers all possible places a bus will transmit from.
- ARC has a number of buildings where the computer hardware can be located and can receive the information from the radio receiver via the Internet.
- The hardware on a bus can transmit a message containing the unique bus ID, timestamp, and location every second. The timestamp and location comes from the GPS receiver on request.
- Historical data for trip times is kept for 24 months.
- The displays have a unique id. They receive information from the central system, not from the buses. There is no communication between bus and display in either direction. There is also no communication from display to central system. The displays just receiving information from the central system and show it.

  The displays should be updated not more than once a minute (more details below).

[1] This is untrue only in that I have had no contact with ARC and so all the details I provide here are completely made up by me. At the time this case study was first created ARC was in the process of developing such a system, which is now operational. In fact, the ARC doesn't exist any more.

Functional Requirements Notes

- The main functionality is that bus patrons will look for information regarding their bus at a given bus stop. This can be thought of as the patrons making a request for the bus arrival information.
- The accuracy of the predicted arrival times is always going to be subject to a number of factors that the system has no control over. For example, missing a green light at an intersection can mean an extra delay of 90 seconds or more, so it is unreasonable to require (for example) that the accuracy be to within the nearest minute. Instead, the "accuracy" requirement is that the estimates shown on the display must be based on data that is not more than 30 seconds old at the point they appear on the display.
- The displays should give the current time, the bus number, and the estimated arrival time for all buses that are scheduled or estimated to arrive within 1 hour of the request.
- If a bus is scheduled to arrive before the time of the request, but is expected to arrive after the time of the request (i.e., it is late), it should still be listed because it is expected arrival time is within an hour.
- If a bus is scheduled to arrive within 50 minutes of the time of the request, but is running 15 minutes late (so the estimated arrival time is over an hour away), it should still be listed because it's scheduled to arrive within an hour.
- The important functionality is to show, at bus stops with displays, the estimated arrival time of buses that stop at that bus stop. There is other functionality that might be desirable, such as:
  - Give the estimated arrival time for the next bus after the time of request on a given bus route at a specific bus stop (e.g., this would be needed to support a web interface or similar).
  - Give the estimated arrival times for all buses that will stop at a given bus stop after a given time.
  - List all the buses that stop at a given bus stop.

Details and Issues

Here are some issues that you may need to consider.

- There are several "non functional" requirements to think about (or consequences of non functional requirements).
  - The main one is performance. There are requirements for getting information to the display in a timely manner (discussed further below). In order to do that, you need to do the estimation in a timely manner, which has processor hardware implications. In order to to that, you need the inputs to the estimation algorithm, which has storage implications. The inputs to the estimation algorithm include (at least) the current bus positions and speeds, so there are communication bandwidth and storage implications. This may not be a complete list.
  - The web site and phone access was mentioned mainly to give an indication that there may be future development. This has implications about some of the quality attributes (e.g., modifiability, reusability).
  - It was never stated anywhere, but there is almost certainly an expectation that this system will last a while (decades rather than years). This has further implications for the quality attributes (e.g., portability).
- If the information a display receives takes more than 1 minute to scroll through, then it will request less than once a minute. If it takes less than one minute, it will just redisplay what it has rather than updating it.

  (I'm assuming there will never be so much information to display that this will cause a problem. Assume there are 40 buses of information to display. The display can show 5 of them at a time. If it scrolls the top one off and the bottom one on every second, it will take 45 seconds for them all to be shown for at least 5 seconds (1 second on each row).)
- You may assume 100 major bus stops, and 1000 buses on the road at any time.

- The phrase “on a regular basis” is meant to be deliberately vague. If the bus is stopped (bus stop or traffic light), then it may not need to transmit every second. But if the bus is travelling 100km/h (whether or not this is legal!), it can do about 30 metres in 1 second, so maybe it needs to transmit more frequently than when it is stopped. You have to decide what to do about this. (Note that the frequency this information is transmitted affects the amount of information the system has to handle, which has implications for bandwidth, storage, and processor speed).

Further Notes

Here are some things to keep in mind.

- You are supposed to be coming up with the *software* architecture for Bus Tracker. You will need to think about the hardware, but don’t spent too long on it — make some reasonable decisions and design your software around it. (A not-unrealistic situation — you’re called in to design software for hardware that’s already purchased…).
- There are going to be some things you won’t know that you need to know to make some of your arguments. For example, what processing power of different kinds of processors have, or how much memory costs. In such cases, make a reasonable assumption and *state it*. (Avoid making assumptions that make your architecture trivial!)