

# BaToo – Enabling the Rapid Prototyping of Mobile Services to Retail Products

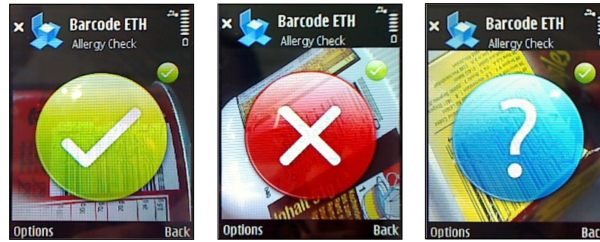
Robert Adelman

Institute for Pervasive Computing, ETH Zurich, Switzerland

**Abstract.** The mobile phone is set to change the way we shop in the future. Consumers will simply scan a product with their mobile phones and access personalized, product-related information and services while they are in the store. To facilitate the interaction with the physical products, consumer-oriented mobile applications require a convenient way to identify the product in the first place, requiring the automated recognition of products. Regarding retail products, a prominent way to do this is to use a mobile phone’s built in camera to recognize the standard 1D barcode present on virtually every product world-wide. The creation of such mobile-phone based applications that provide services and information to real-world objects is currently very attractive: For prototypes, technology demos, or user studies. But even though many components of applications and prototypes are recurring, the creation of even simple applications requires a lot of time as well as know-how – both limiting the progress and development of new mobile-phone based applications and ideas. This document outlines our contribution to this problem: A rapid prototyping platform that includes the robust recognition of 1D barcodes and allows the creation of easy to use mobile-phone based applications to retail products: Within minutes, without any know-how about mobile phone programming.

## 1 Introduction

Today’s consumer goods packaging lists a significant amount of product-related information. This includes nutritional information, ingredients, and possibly handling or recycling instructions. Some product packaging also comprise promotions with links to free song downloads or competitions. Due to the limited amount of space available on the product packaging and its static nature, the information cannot be customized for each consumer. Visually impaired people might prefer seeing allergy-related information in large print and non-natives might like to see the information in a different language. There is also a wealth of additional product-related information available that is not directly printed on the product packaging at all due to size constraints and possibly commercial considerations, e.g. reviews by consumer watch groups or price comparisons. Mobile phones have the potential to address many of these issues since they comprise display, long-range communication capabilities, processing, and user profile storage capabilities [2]. Since many of these potential applications are



**Fig. 1.** Screen-shots of the Allergy-Check application

especially useful when being "on the go", e.g., while shopping, a simple and fast user interaction is essential, requiring the automated recognition of objects. Even though RFID technology is very promising, the widespread use of RFID tags on retail products remains unlikely for the next years. In contrast, barcodes are ubiquitous - printed on virtually all consumer items world-wide.

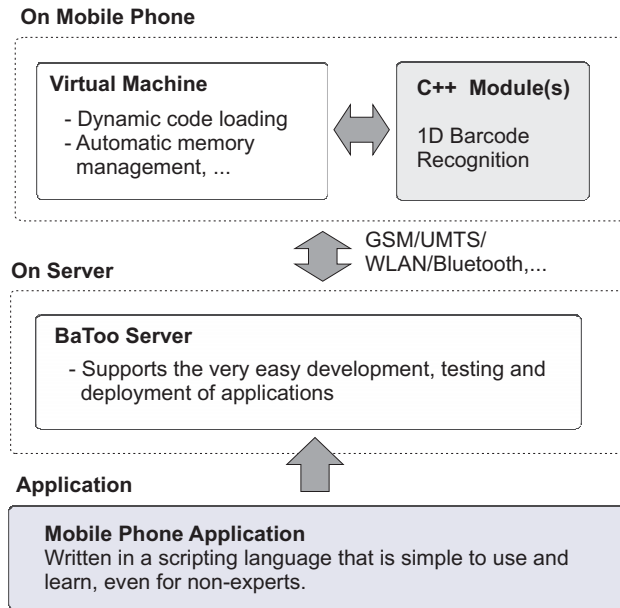
On one hand there is an abundance of potentially highly useful applications for both consumers as well as companies, on the other hand implementing applications ideas or prototypes is very difficult. This is due to the required know-how for the optical code recognition as well as the often necessary time consuming and difficult C++ Symbian programming on devices. In this document we are presenting both the outline of a rapid prototyping platform that eases and accelerates the development of according mobile phone applications, as well as the underlying powerful recognition of 1D barcodes on standard mobile phones, using the built-in camera.

## 2 Applications

Figure 1 shows screenshots of a typical application to retail products: the Allergy-Check application. It is based on the recognition of 1D barcodes. Once the user defined a profile containing all substances he or she is allergic to, holding the mobile phone in front of a product's barcode gives the user a simple answer to the question "Is that product fine for me?".

## 3 Rapid Prototyping Platform

Like already mentioned, developing mobile phone applications that require the recognition of real-world objects is, due to several reasons, often a very time consuming and tedious process. The goal of our rapid prototyping platform is to foster the creation of novel and innovative applications by enable non-professional programmers to create mobile phone based services to real-world objects, specifically retail products. Our emphasis is hereby not on replacing, but on complementing existing general programming environment for mobile phones, such as J2ME or C++ Symbian with features that enable a very easy and rapid creation



**Fig. 2.** General architecture of the prototyping platform

of prototypes, without requiring any knowledge of mobile-phone programming or the involved technologies.

The general architecture of the platform is shown in Figure 2: The mobile phone contains predefined C++ Symbian components (libraries) for the major tasks that need the full processing power and capabilities available on the phone, mainly the 1D barcode recognition. Given these components, experienced C++ Symbian developers can already create functioning applications, even without special knowledge – for example regarding image recognition. But implementing even simple C++ Symbian applications still takes a considerable amount of time and won't be feasible for inexperienced developers – due to many Symbian specific concepts like ActiveObjects or Descriptors, complex memory management or the lack of documentation. J2ME is more accessible, but often lacks the required APIs and speed. In order to allow the simple and fast creation of applications, the BaToo system allows the user to write applications in a very easy to learn and use scripting language that is then executed on a virtual machine on the mobile phone, which has access to the provided C++ modules.

An additional BaToo-server-application running on a remote desktop computer handles the remaining complexity involved in developing applications: For example testing the application on the mobile phone, or creating distributable applications for today's Symbian based mobile phones, including the packaging and mandatory signing of applications. This way, developing a new service to

