'MichelinCook' - an easy way to cook better

Pratheek LK
Department of Computer Science and Computer Engineering
Texas A&M University

Srinath Nadimpalli
Department of Computer Science and Computer Engineering
Texas A&M University

Sri Jeevanige
Department of Computer Science and Computer Engineering
Texas A&M University

Smit Shah
Department of Computer Science and Computer Engineering
Texas A&M University

Jennifer Ma
Department of Computer Science and Computer Engineering
Texas A&M University

ABSTRACT
In this paper, we describe a novel method for delivering cooking instructions in the form of self-timed digital cards. We will discuss the shortcomings of current available methods for providing cooking instructions to the user. The merits of self-timed timed digital cards presented in the mobile application MichelinCook is then discussed. We will then discuss the design and implementation of MichelinCook as an android application. The paper also describes the design of hardware system for controlling the temperature of cooktop and integrating it with the mobile application. Finally we discuss the evaluation methods for the system and about the scope for future work.

Keywords
Self-time digital cards, recipe delivery, mobile application, temperature control

1. INTRODUCTION
The kitchen plays an important emotional role in our lives. Everyone enjoys getting together with friends or family for meals, especially during festivals or holidays. These celebrations are usually potluck in nature and it would be awkward to bring in off the shelf food for such meet-ups. Moreover, it is believed widely that âĂIJThe easiest way to ManâĂŹs heart is through his stomachâĂİ, making it more important to cook for your loved ones. It also becomes essential to cook your own meal when you are on a particular diet or want to have a healthier lifestyle. So it is no exaggeration to say that ability to cook improves your life greatly.

But there are many people who do not take to cooking because of the fear that manifests from this exercise. Some people are overwhelmed by the recipes which may be complicated to cook from or overly long and fear missing the steps, some are intimidated by the cooking techniques involved because of not clearly understanding them, some fear that they may overcook the food, some fear seasoning the food wrongly and there are some others who worry about how the food looks. This phobia or fear of cooking is termed as Mageirocophobia.

However we believe that it is possible to alleviate the above mentioned fear by providing an easy and intuitive user interface to deliver the cooking instructions.

2. PROBLEM STATEMENT
The current available system for delivery of cooking instructions is difficult to follow and confusing to cook from. We did a survey to find out what are the different medias that people use to cook from and found out that people use recipe blogs, YouTube videos, paperback recipe books, recipe apps etc..,

There are several shortcomings of using the above listed medias for cooking. The conventional paperback and recipe blogs are difficult to cook from because of them being static and less intuitive. Using YouTube videos during cooking is cumbersome as it is difficult when trying to refer to a previous step and results in a lot of pause, rewind and forward. The current available apps provides solution to many of the above mentioned problems but we believe that the cooking experience can be further enhanced by feature set of MichelinCook.

Our mobile application mainly aims at providing the user with cooking instructions in the form of self-timed digital cards. Flash Cards have been used as an effective strategy for studying, we use a modified version of the flash cards to make up the cards used to provide the cooking directions. The cooking procedure is broken down into many steps and each card presents an individual step with a timer(which runs down from the required time for that step). The important steps are additionally presented with photographs on the cards and can be used as references to further enhance the cooking process.

3. DESIGN

The design goal of MichelinCook is to provide timing based delivery of flash cards containing cooking directions. Following are the important design goals of the application.

1. Easy search and access to Recipes
2. Access to latest trending recipes and curated recipes from popular chefs
3. Finding recipes based on categories like 'Cuisines, 'Quick & Easy', 'Cheap Eats' and 'Healthy'
4. Direct search for a recipe from every page
5. Easy navigation throughout the app

Interactive content delivery for cooking Table of required quantity of ingredients for selected recipe The quantity of ingredients are appropriately changed depending on the number of servings selected Cook-top size is considered to scale the time required for each cooking step Timer enabled flash cards containing cooking direction and picture to assist while cooking Ability to navigate to any step of the cooking procedure The above listed features will help provide user with an easy and direct way to cook from. The overall experience is further enhanced by a simple UI design.

4. SOCIAL AND COGNITIVE ISSUES

It should go without saying that smartphones make our lives easier in many ways. They help us find directions to our next destination, maintain communication with loved ones and dear friends, and even waste precious time playing games. We carry our phones with us everywhere we go, be it our living room, work space, or even the bathroom. So why shouldnâĂŹt we venture with it into the kitchen? Cooking apps are the next step in the complete mobile takeover.

The app revolution is leaving no sphere of life untouched, not even the kitchen. With more devices featuring large and vivid touch screens, it doesn't take a genius to realize that a good tablet with the right apps can make for a handy digital cookbook, able to help home cooks and pro chefs discover and organize recipes, track shopping lists and organize menus and meal calendars. From recipe discovery apps and classic cookbooks, to personal recipe organizers, there are now apps for Android, iOS, Windows Phone and Windows 8 devices.

But with MichelinCook user experiences features that are clear and concise. The digital cards provides a unique solution to enable users to cook a variety of dishes. The digital cards are designed for ease of learning, with step-by-step instructions, allowing users to see exactly what's going on all the time.

5. IMPLEMENTATION

5.1 Step 1 : Application Workflow and UI Design

he application was conceptualized and prototyped using Justinmind prototyper. The screen shots of the prototype is as shown below,

Figure 1: Main Screen

Figure 2: Navigation Drawer to access the different categories of menus

5.2 Step 2 : Android Implementation

5.3 Step 3: Database Implementation

We have used android SQLite database for this application. SQLite is a very light weight database which comes with Android OS. Basic Database design for the application is as follows 10.

1. Cooking_Category : This is a master table for recipe categories based on cooking method. It contains category id and name where id is the primary key.
2. Regional_Category : This is a master table for recipe categories based on regional origin of the recipe. It contains regional category id and name where id is the primary key.
3. Other_Category : This is a master table for other

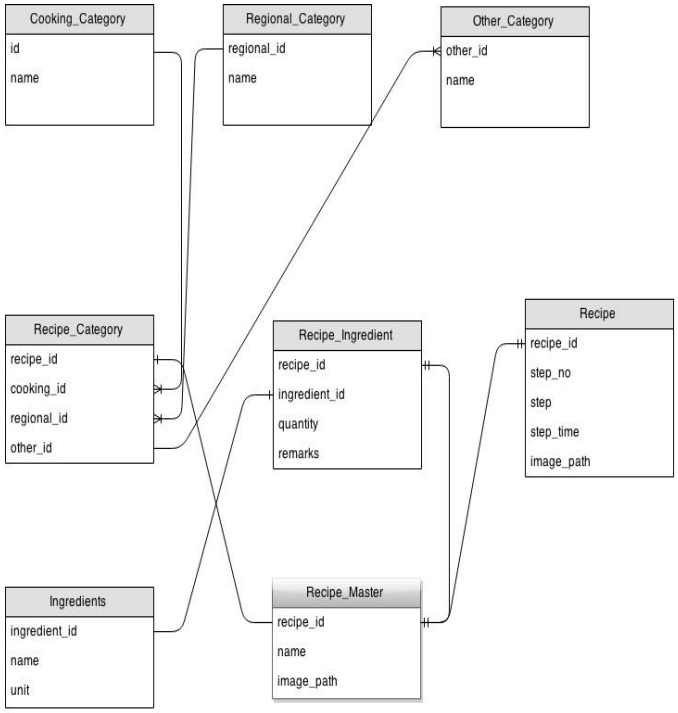

Figure 10: Database scheme

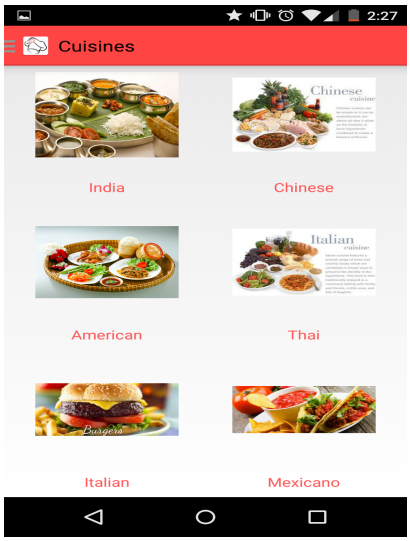

Figure 3: Origin based categorization of cuisines

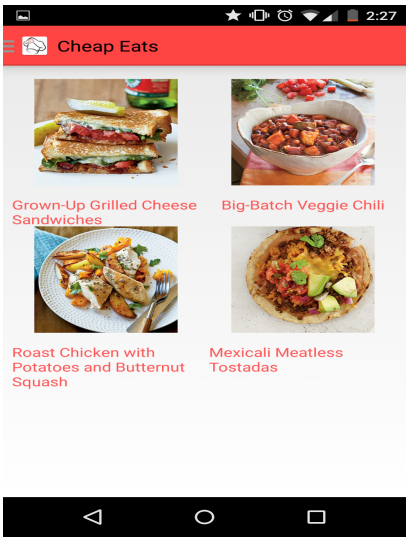

Figure 4: Different categories of recipes - Cheap Eats

recipe categories such as Quick & Easy, Cheap eats or Healthy Recipes. It contains category id and name where id is the primary key.

4. Ingredients : This table contains all the ingredients for all of the recipes inside database. This table contains id, name and quantity unit of the ingredient where id is the primary key.

5. Recipe_Master : Recipe_Master table is master table where all the recipes, their ids and display image paths are stored. Recipe_id is the primary key of this table.

6. Recipe_Category : This table contains information about which recipe belongs to which category. It contains recipe_id, cooking_id, regional_id and other_id which all are Foreign keys of Recipe_Master, Cooking_Category, Regional_Category and Other_Category tables respectively.

7. Recipe_Ingredients : This table contains list of ingredients for each recipe along with quantity and any other remarks. recipe_id and ingredient_id are the foreign

Figure 5: Different categories of recipes - Healthy

Figure 6: Different categories of recipes - Quick & Easy

keys from Recipe_Master and Ingredients tables respectively.

8. Recipe : This table contains all the steps required to perform cooking with step number, step time and image (if available) for each step. Recipe_id is the foreign key from Recipe_Master table.

5.4 Database Integration with the Application

Once the database is implemented, we integrated our application with it using Android SQLite APIs. We populated our user interface with data read from Database using Database Helper Class .In Database Helper class, we have implemented different helper methods that execute SQL queries using rawQuery API of SQLite and retrieve the appropriate recipe data and pass it on our Android activities. Recipe categories implemented as Android Fragments such as Cuisines,

Figure 7: Recipe Review screen that shows the ingredients

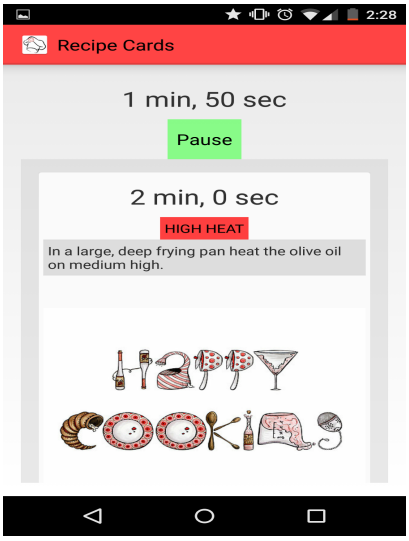

Figure 8: Recipe card that enables easy delivery of cooking instructions

Quick and Easy, Cheap Eats, Healthy calls their corresponding Database Helper methods. They return data back to these Categories. Once the recipe category fragments receive data from database helper methods they show the content such as recipe text and recipe image to the user. When the user clicks one of the dishes, The recipe category Fragment passes the Recipe ID as part of intent object to the Recipe Preview screen activity. Recipe preview screen activity after retrieving the Recipe ID from the intent object calls Database Helper method with the Recipe ID to retrieve the appropriate list of ingredients and their quantities and Recipe Image on the Recipe preview screen. After the user reviews the recipe review screen and decides to cook that dish, they will click the Cook button. When that happens, the same intent bounded with Recipe ID will be passed to Recipe Digital Card activity. Similar to the

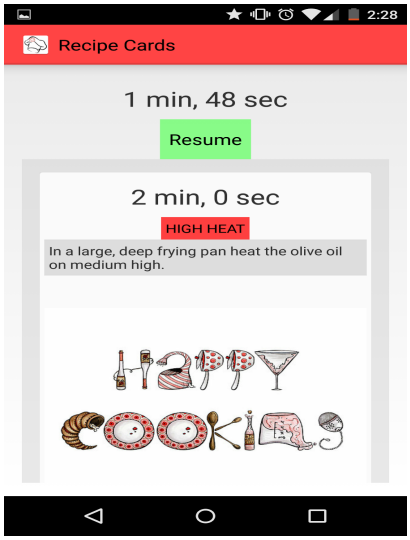

Figure 9: Recipe card that enables easy delivery of cooking instructions

Recipe preview screen, Recipe Digital card activity retrieves List of recipe steps, time required for each of the recipe steps, Images required for each of the recipe steps from the Database Helper class by passing on the Recipe ID . Essentially, Database helper uses Recipe ID and executes the SQL query on database to get the corresponding results.

5.5 Step 4: Hardware System Design

An embedded system made up of a cook-top, a microcontroller with a Temperature sensor and a Bluetooth module.

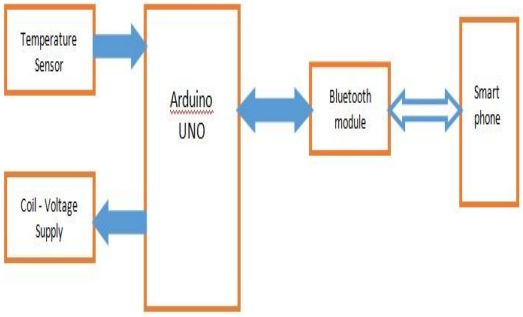

Figure 11: Top Level design of Smart Cook-Top

The temperature sensor senses the current temperature of the coil and then sends this data in the form of an analog signal(voltage signal 10mv/âĎČ). The arduino reads the voltage level on the analog input pin available on the board. The voltage level is processed in the arduino and the equivalent temperature information is sent on the serial communication lines to the bluetooth module, this temperature information is then wirelessly transmitted to the paired smartphone. The temperature value can be viewed on MichelinCook.

The desired temperature value of the cook-top is sent from

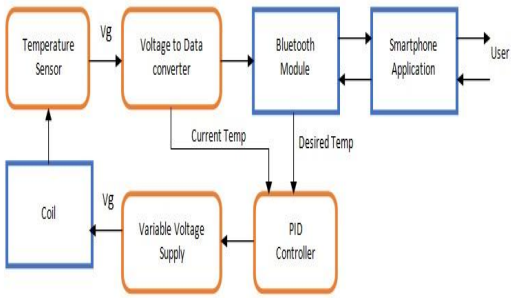

Figure 12: Data flow model of Smart Cook-Top design

the MichelinCook application to the bluetooth module which is then sent to the arduino module. To set the cook-top to a particular temperature, a PID algorithm is implemented in the arduino, it makes use of the current temperature and the desired temperature values to calculate the required change in the voltage value. The new voltage is supplied to change the temperature in steps till the desired value of temperature is reached.

The cook-top to communicate with MichelinCook was prototyped in simulink and the simulink model was used to simulate the design to check for its working and functionality.

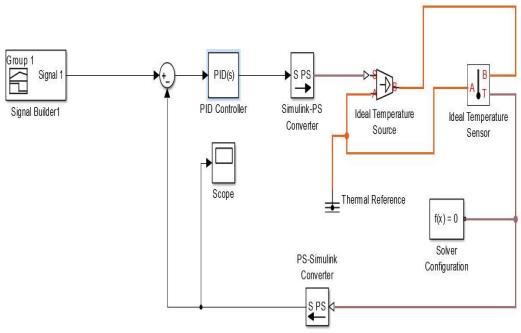

Figure 13: Data flow model of Smart Cook-Top design

The signal builder is configured to present temperature values relative to time. The signal builder has temperature values of 80C from 0s to 300s and 100C from 300s to 1000s. The cook-top coil is modelled as an Ideal temperature source in this phase of the project. The closed system has a PID controller which is a commonly used controller in industries.

The PID values are selected to minimize the rise and have low overshoot as possible. The output of the closed loop system is monitored using the scope. The signal waveform corresponding to input signal from signal builder and output temperature as sensed by the temperature sensor is as shown in figure below.

To develop the mathematical response for the real [non ideal]

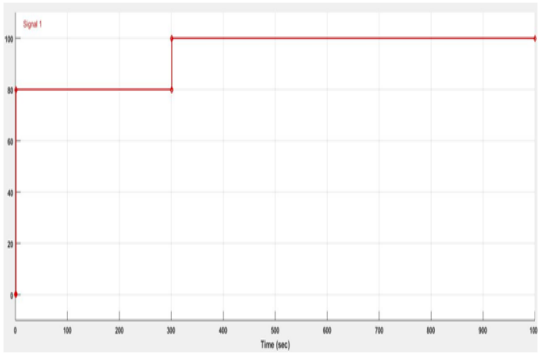

Figure 14:

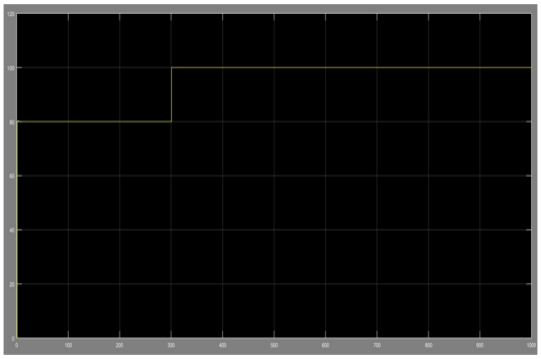

Figure 15:

cook-top process is very important. Most processes similar to cook-topâĂŹs are modelled using First or Second order plus dead time models.

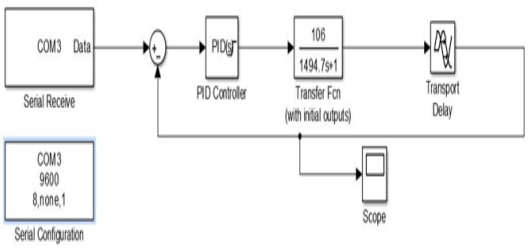

Figure 16: Real Cook-Top Design with real cook-top behavior modelled using FODPT

The response graph for the real cook-top is as shown in the figure below.

6. EVALUATION

We will use Lean Startup model for evaluation of our mobile application idea. The advantage of using Lean Startup Methodology in mobile development includes,

- Faster time to market

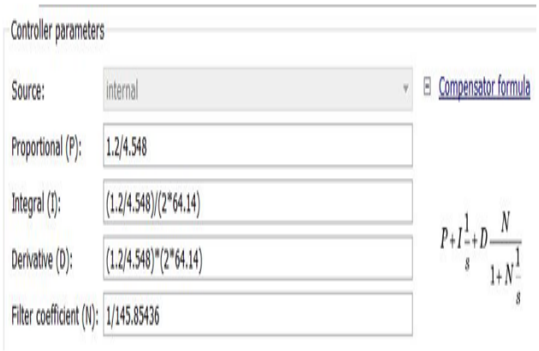

Figure 17: PID parameters

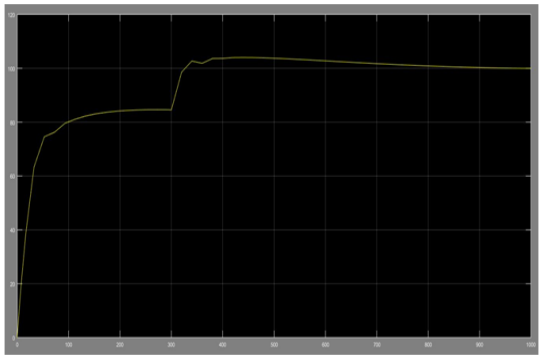

Figure 18: Temperature response of real(FODPT) cook-top coil

- Lower Cost Overall and Upfront Cost
- Less Financial Risk
- Build a Better product that customers want.
- Less wasted time

In our case as this is a course project, the most relevant use of Lean Startup Methodology is for building a better product that customer wants and with a shorter design cycle time.

6.1 Building an MVP

A minimum viable product is a product with only the core features. We plan to use the current version of our application as MVP for the purpose of testing. The MVP will be deployed for testing among early adopters whose feedback will be collected for further enhancement of the application.

The MVP is a way to test with the objective of answering four questions:

- Do users recognize that they have the problem we are trying to solve?
- If there is a solution, would they use it?
- Would they use our app?
- Are we building a solution for the above problem?

If the answer is yes for four of the above questions, then we have an actual requirement for such an application. Building an MVP is a way to collect feedback, user response, and test the waters. The challenge of building an MVP can be a bit more difficult on mobile. ThereâĂŹs a higher bar for mobile applications. We have to be a bit more refined as people are less patient on mobile and thereâĂŹs a much lower tolerance for bugs.

6.2 User Feedback

Each feature built are like experiments to see how users adopt and engage in the product. In the early stage of building a product, itâĂŹs extremely important to talk to your customers. The early adopters, beta testers, and evangelists are key for customer discovery. We need to evaluate if our core proposed product features solve their problem? Data gathered from the first group of beta users will be key for figuring out what features to focus on.

6.3 Quantitative and Qualitative Metrics

The feedback from users are extremely useful and comes in both qualitative and quantitative forms. There are certain key metrics to focus on that will demonstrate health. One of the most important focus for our application is to measure engagement. Nothing signals stickiness and that you have something with your MVP than to have good engagement in your app. Some hard metrics that shows engagement in mobile includes:

- Active Users âĂŞ Number of active users a day/week/month is essential for measuring
- Session Length âĂŞ How long do the users spend in your app. The time they open to the time its closed.
- Session Interval âĂŞ How long in between each session of usage
- Screen Flow âĂŞ Interactions in the app between each screens, duration on each screens, and the total number of occurrences in each screen.
- Retention Rate âĂŞ the percentage of users who return to the app based on the date of their first visit.

In addition to these quantitative forms of measuring engagement, itâĂŹs also important to observe and watch users engage in our app for UX purposes. Direct communication from users may be best for clear feedback of the app.

Any user data points gathered, both qualitative and quantitative, can be extremely beneficial for a glance at user behavior to learn and provide direction for app development. It is crucial to track user behavior, feedback, and use the information when building new features.

7. FUTURE WORK

In this section, we discuss our future work from following perspectives: implementation, functionality and user experience.

7.1 Implementation

Implementation: Currently we only implement âĂIJMichelinCookâĂİ on Android system. In the future, we also plan to implement it on iOS, which is also one of the most popular smartphone system. And we also plan to upload our app to both Android Market and iOS App Market.

7.2 Functionality

Functionality: self-timed digital card is very helpful for cooking, however, it needs users to manually click each card. In the future, we may add voice control so that users can cook the dish without frequently clicking their smartphone. In addition, in the future, we plan to add functions to allow the user to upload their own recipes and some happen-before relations (e.g., step 1 should go first than another step 2) for their recipes , then our app can automatically generate the optimized step sequence with corresponding time card.

7.3 User experience

User experience: In the future, we plan to make a survey of our app to other students and ask some feedback about current interface, functions and features. In this way, we can add more new functions and features based their feedback.

8. CONCLUSION