Adarza Technologies
Simon Fraser University
Burnaby, BC, V5A 1S6
Email: info@adarza.com

February 22, 2005

Lakshman One
School of Engineering Science
Simon Fraser University
Burnaby, British Columbia
V5A 1S6

RE: ENSC 440 Functional Specifications for the Bacteria Classification Assistant

Dear Mr. One,

The attached document, Functional Specifications for the Bacteria Classification Assistant, outlines the features that will be available upon completion of our project. Our system will significantly increase the rate by which microbiologists identify bacteria colonies, resulting in substantial savings in time and manpower.

From the feedback collected through meetings with our client, we have generated a set of requirements that our system must meet. Additional features that the user may find useful have also been included. This document will also consider the limitations and future evolution of our software.

Please feel free to contact us if you have any questions, comments, or concerns about this document. We can be reached by e-mail at *info@adarza.com* or by phone at 604-437-6599.

Sincerely,

Edward Goh

Edward Goh
President and CEO
Adarza Technologies

Enclosure: Functional Specifications for the Bacteria Classification Assistant

Adarza Technologies
Functional Specifications for the Bacteria Classification Assistant

Project Team:
Edward Goh
Geeda Ip
Josna Rao
Linda Wu

Contact Information:
Edward Goh
info@adarza.com

Submitted to:
Lakshman One
Mike Sjoerdsma
Steve Whitmore
Scott Logie
Tony Ottaviani
Amir Masoud Niroumand

Date Issued:
February 22, 2005
Revision 1.11

Executive Summary

Our discussions with the employees at the British Columbia Center for Disease Control (BCCDC) indicate that currently lab technicians identify bacterial colonies by examining the samples under a microscope and applying their personal knowledge [1]. However, rare strains of bacteria with unique visual characteristics are often difficult to commit to memory, and extensive database searches are required to correctly identify the sample. Our system does not attempt to replace the technician or microbiologist. Instead, our system will significantly reduce the number of database entries that must be processed manually.

The BCCDC currently possesses a Canon 5 mega-pixel camera connected to a dissecting microscope via a camera adapter. The Bacteria Classification Assistant (BCA) will be optimized for similar hardware setups but if suitable image processing is performed prior to introducing the picture to the software, lower resolution cameras may be used.

We have identified four major visual criteria for bacterial colony classification: colony size, overall colony colour, overall colony shape, and colony boundary structure. Three modules within our software will compute these characteristics. After the user has input an image file of the sample, the BCA will process the image and search its internal database to generate a list of likely candidates. Further classification criteria may be added in order to narrow down the search, but we do not expect the BCA to be able to correctly identify more than the genus of the bacteria. The user will be able to add, delete, and modify entries in the database to allow for future identification of additional strains.

For our proof of concept demonstration, we will require approximately fifteen to twenty pictures of samples from the BCCDC in order to populate a database. An addition ten different samples that comprise a subset of the database will be used to verify the performance of the BCA.

According to our time schedule, we expect to have the software integrated with the database and ready for demonstration by the third week of April.

Table of Contents

List of Figures

List of Tables

Glossary

BCCDC	British Columbia Center for Disease Control
BCA	Bacteria Classification System
CC	Classification Criterion

1 Introduction

The Bacteria Classification Assistant (BCA) is a program that allows the user to identify unknown strains of bacteria from images of the bacterial colony taken through a microscope. The BCA will have the user input the colony image and a few parameters into the program. By processing the image, the program will isolate four key properties of the colony: area of the colony, colour, colony shape characteristics and colony edge characteristics. Once this data has been collected, the BCA will compare the data to a stored database and will return possible candidates. The database of stored images will have functions to allow the user to search for specific characteristics via a text input and will be expandable by the user.

1.1 Scope

This document details the basic specifications for a proof-of-concept version of the BCA. Additional specifications regarding enhanced features of our system are also described and will be completed time permitting. Product constraints, system evolution, test plans, and descriptions of user documentation are also included. We intend to use this document as a measure of our success in designing the BCA.

1.2 Objectives

The following notation will be used throughout this documentation to describe our specifications:

[S#-x] Functional Specification

− Specification number
x – Priority of the requirement
H – High Priority Requirement
L – Low Priority Requirement

2 System Overview

The bacterial classification system will aid the researcher in identifying unknown strains of bacteria colonies from an image taken by a digital camera microscope. Once all the information is entered, the program will begin to compile a list of possible identifications from its database, listed by their probability of matching the unknown specimen. This program will eliminate a significant portion of candidates using the visual criteria of colour, size, colony shape, and colony edge characteristics. The four criteria will be individually processed by program modules and will return a category for each characteristic. The collected data on each unknown sample will then be processed in a decision module to generate a list of possibilities. The possible candidates will be displayed and the researcher can then make further determination in order to identify the unknown colony. The functional block diagram of the BCA is shown in Figure 1.

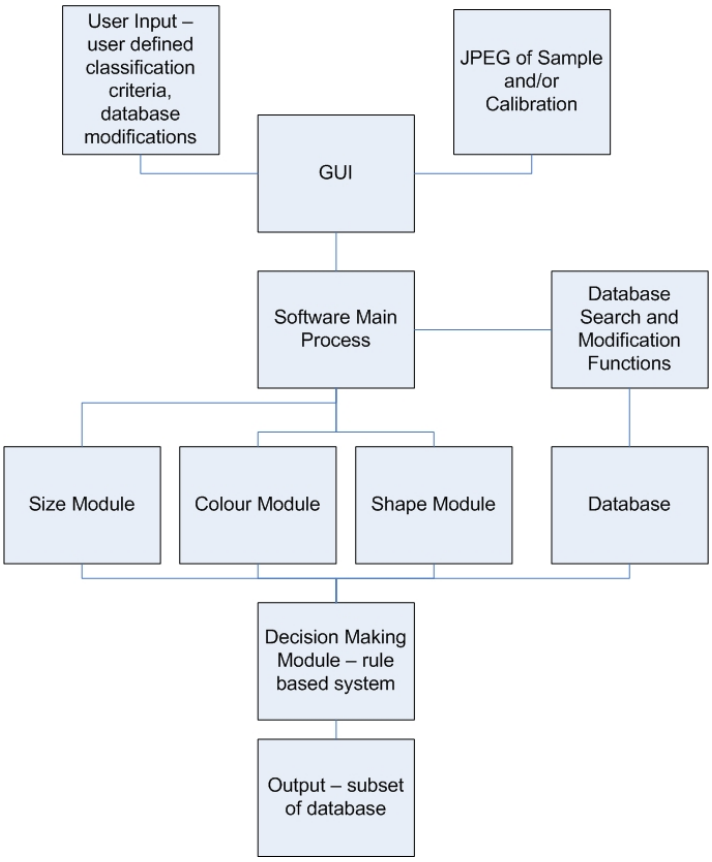


Figure 1: Functional Block Diagram of the BCA

Our program will also include a database of known and characterized bacterial colonies from which the list of possibilities will be generated. The database will also have basic searching tools to isolate strains by individual characteristics, expanding our program from just being a valuable tool for identification.

3 Input Requirements

The software must receive certain information in a specific picture format in order to be able to match the samples with entries in the database.

[S1-H] The software will work with 24-bit colour JPEG files with a resolution of at least 640x480 pixels.

[S2-H] The software will require a calibration picture for each sample.

[S3-L] The software shall receive all picture formats (e.g. BMP, PNG, GIF, TIF) as input.

4 Size Module Requirements

The size module is responsible for determining the area covered by the bacteria in relation to the calibration image.

[S4-H] The size module will classify the size of valid samples, with 90% accuracy, to be one of four possible output constants: large, medium, small, or tiny.

[S5-L] The size module shall obtain the calibration image automatically from the database.

[S6-L] The size module shall compute the percentage difference between the size of the calibration sample and the size of the test sample.

5 Colour Module Requirements

The colour module will take an input image and categorize it into the following categories: white, grey, or strong pigmentation. Although there are many colour variations in bacteria colonies, researchers use the most dominant colour as an identification criteria. Therefore, a broad category such as “white” is sufficiently accurate. Strong pigmentation refers to colours like blue, green, red, etc. The hues of each colour occurring within the colony are irrelevant for identification purposes and will not be separately analyzed by the colour module.

[S7-H] The colour module will return a parameter for one of the following categories: white, grey, or pigmentation. Pigmentation will be divided into red, blue, green and yellow categories. The parameter returned will be accurate 90% of the time.

[S8-L] The module shall analyze colour variation of bacterial colony and return a parameter for one of the following categories: mottled, centre bright spot, and transparent.

6 Shape Module Requirements

The shape module will consist of two sub-modules: colony shape, and boundary shape.

6.1 Colony Shape Sub-Module

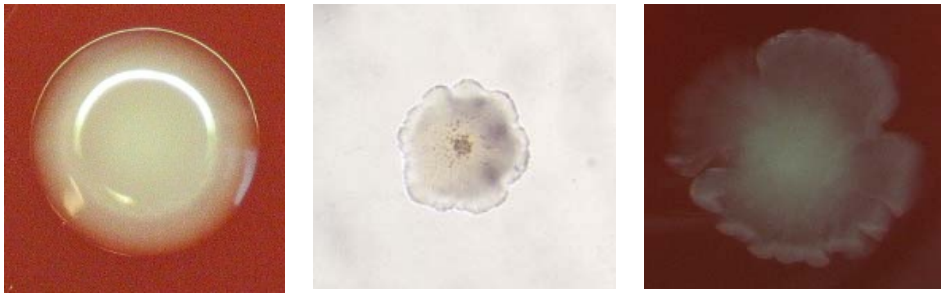
The colony shape sub-module will take the input image and classify the general shape of the colony into two categories: circular or irregular. This general shape does not include the contour features, i.e. if the edge is undulating or smooth. A specific degree of curvature will be specified in order for a bacterium to be considered circular or not.

[S9-H] The colony shape module will return whether the colony shape is circular or irregular with 85% accuracy.

[S10-L] The module shall return a percentage corresponding to the level of certainty for the chosen shape category.

6.2 Boundary Shape Sub-Module

The boundary shape module will take the input image and classify the specific edge feature of the colony into three categories: smooth, undulating, or finger-like. A smooth edge will have a relatively constant angle between the tangent vector and the normal of the radius vector (this angle will be called the edge angle). An undulating edge will have small rises and troughs in the edge, so that there will be a slight variation in the edge angle. A finger-like edge will have large rises and troughs and may fold in on itself, so that the edge angle may be greater than 90°. Examples of these categories are shown in Figure 2.



(a) Smooth (b) Undulating (c) Finger-like

Figure 2: Examples of Smooth, Undulating, and Finger-like Edges

[S11-H] The boundary shape module will, with 80% accuracy, return one of the following boundary shape categories: smooth, undulating, and finger-like.

[S12-L] The module shall return a percentage describing the level of certainty of the chosen boundary shape category and will return multiple boundary shape categories.

7 Output Module Requirements

The output module is responsible for collecting all the data generated by each of the three classification criteria modules and producing a list of possible bacteria strains for the researcher to further eliminate.

[S13-H] The output module will return a set of possible candidates. The size of this set will be at most half the size of the current database.

[S14-H] The generated list will include additional data for the possible candidates, taken from the database.

[S15-L] The generated list shall be ordered by the probability of the match to the unknown sample.

8 GUI Specifications

The GUI allows the user to manipulate the database, specify input images and parameters, and extract information from the results given by the output from the software.

[S16-H] The GUI will allow the user to input sample images and prompt for relevant data to be entered.

[S17-H] The GUI will display the results of image analysis in a list with links to their database entries.

[S18-H] The GUI will allow the user to add and remove known entries from the database. Each addition and removal will be confirmed by the user first. For additions, the GUI will also prompt the user to enter the relevant CC's.

[S19-H] The GUI will allow the user to browse the database entries and access all information stored in each entry.

[S20-L] The GUI will allow the user to perform advanced search options by CC's through the database entries.

9 Performance Requirements

9.1 Response Times

[S21-H] The BCA will load a user selected bacteria image file to the GUI in less than 3 seconds on a computer with a 1.5GHz processor.

[S22-H] The BCA will analyze the bacteria image file and display possible candidates in less than 30 seconds using a database of 300 records on a computer with a 1.5GHz processor.

9.2 File Capacity

Table 1 lists the storage requirements for a typical BCA database. The record sizes for each table are given, as well as estimates of the typical number of records in each table. Note that image file sizes are not included in the calculation.

Table 1: Estimated storage requirements for the BCA database (excluding image files)

Table Name	Size Per Record (Bytes)	Typical Number of Records	Total Size (Bytes)
BacteriaInfoTable	202	300	60,600
LightConditionTable	104	20	2,080
PhotoTable	109	350	38,150
SizeTable	39	4	156
ShapeTable	55	10	550
Total:		1084	101,536

Each PhotoTable record will reference a JPEG file with an average size of 4 MB. Since there are typically 350 records in the PhotoTable, the storage requirement for the JPEG files will be 4 MB * 350 = 1400 MB.

[S23-H] The total storage for the BCA data will be approximately 1410 MB.

[S24-H] The size of the BCA executable program will be less than 1 MB.

10 Host Computer System Requirements

The BCA will also have certain requirements for the user's computer on which the program will be run [2].

[S25-H] The processor will be at least a 90-megahertz (MHz) Intel Pentium-class processor, or an AMD Opteron, AMD Athlon64 or AMD Athlon XP processor.

[S26-H] The system will have at least 96 MB of RAM and 1.5 GB of hard drive space.

[S27-H] The system will be running one of the following operating systems: Windows XP Professional, Windows XP Home Edition, Windows 2000, Windows Millennium Edition, Windows 98, or Microsoft Windows NT 4.0 Service Pack 6a.

[S28-H] The system will have Microsoft Access 2000 (or later), Microsoft Data Access Components 2.6, and Microsoft .NET Framework 1.1 Redistributable installed.

11 Testing

The BCA will undergo rigorous testing during and after the software development phase.

11.1 Testing During Software Development

To ensure requirements conformity, the BCA will be written using a test-driven developmental process [3].

[S29-H] Unit testing will be carried out using the NUnit framework [4].

[S30-H] Test suites for business rules and functionality will be written based on each module's requirement specifications; as new code is written, the entire program must past all existing tests before the new code can be added.

11.2 Database Tests

To ensure the database is behaving as it should, the following tests will be conducted:

[S31-H] Tests of sample pictures will be run in the following cases: sample is not in database, sample has exactly one match in the database, and sample has no matches in the database.

[S32-H] Addition of new records will be tested on the follow database states: empty, populated (but not full), and full.

12 Documentation and User Training

To ensure that users of the BCA become fully competent with the program, Adarza will provide onsite user training, user manuals, and maintain an up-to-date website.

[S33-H] All documentation and user training will assume user has basic knowledge of digital cameras and Windows OS, and professional knowledge of microbiology.

12.1 User Manual and Website

[S34-H] The website and user manual will be written in Canadian English.

[S35-H] Minimum system requirements for the installation of the BCA will be clearly listed on the website and in the user manual.

[S36-L] Step-by-step instructions on the installation of the BCA, on using the BCA to identify the genus from a picture of bacteria, and on adding new bacteria to the BCA's database will be provided

[S37-L] The user manual shall include a Frequently Asked Questions (FAQ) and troubleshooting section.

[S38-L] The website shall include instructions and examples on hardware setups for capturing digital images.

12.2 Onsite Training

[S39-L] A member of the Adarza training team shall provide a one-hour, onsite training session for each site license of the BCA.

[S40-L] The training session shall include demonstrations of using the BCA to identify the genus of a bacterial image and making modifications to the BCA's database.

13 Constraints

The BCA will have the following hardware, software, and database constraints:

13.1 Hardware Constraints

[S41-H] The BCA will not be run on a non-IBM compatible PC.

13.2 Software Constraints

[S42-H] The BCA will not be run on non-Windows based systems.

[S43-H] The BCA database files will reside on the local hard drive. The database cannot be accessed over a network.

13.3 Database Constraints

The database will be stored in Microsoft Access. Therefore, the database of the BCA will be subject to the same constraints as Microsoft Access [5]:

[S44-H] Maximum database file size will be 2 GB minus the space needed for system objects.

[S45-H] Maximum number of objects in a database will be 32,768

14 System Evolution

In the future, additional features for the BCA can be implemented to create a more robust system.

[S46-L] The BCA shall identify other organisms such as parasites.

[S47-L] Additional user-specified classification criteria shall be implemented.

[S48-L] The database shall be stored on a central server to allow for multiple users and simultaneous access.

15 Conclusion

The functional specifications described in this document are the expectations that Adarza Technologies has of the BCA in terms of the internal module requirements, the user documentation requirements, and the overall performance requirements. In addition, the constraints and limitations that are expected of the BCA are included, along with future features. Adarza Technologies intends to use this document as a guideline when developing and testing the BCA in order to create a viable product capable of competing in real world markets.

References

[1] Dr. SH Goh, (private communication), February 2005.

[2] Microsoft Corporation, ".NET Framework 1.1 Redistributable System Requirements", Feb. 2005; http://msdn.microsoft.com/netframework/technologyinfo/sysreqs/default.aspx.

[3] Agile Alliance, Feb. 2005; http://www.agilealliance.org/home.

[4] J.W. Newkirk and A.A. Vorontsov, "NUnit", Feb. 2005; http://www.nunit.org/.

[5] Microsoft Corporation, "Access Specifications", Feb. 2005; http://office.microsoft.com/en-ca/assistance/HP051868081033.aspx.