The Development of an Accessible Web-based Quantum Transportation Device for Older Users

| **Matilda F. Bloggs**  **Final Year Project**  **BSc (Hons) Applied Computing**  **University of Dundee, 2005**  **Supervisor: Prof. A. N. Oldhand** |
| --- |

***Abstract* –** The potato plant is the fourth most grown crop in the world and therefor a significant portion of the world’s food production and feeding basis for cattle. There are more than one hundred known diseases and pests which threaten the potato plant and have had dramatic consequences in the past. The James Hutton Institute in Dundee, Scotland, has asked us to design a diagnostics application for smartphones which enables a unexperienced and/or unskilled user to identify pests and diseases in both the tuber, leaves and stems, with appropriate guidance and troubleshooting upon identification. The project case study focuses on farmers in Malawi. The application is therefore aimed to help those farmers specifically. This includes catering to internet access standards, smartphone types and language barriers. The finished product fulfils all requirements and enables non-technical access to the application database as well as an intuitive interface and concise help upon identification. *The abstract (or ‘executive summary’) is an important part of your report. In essence, it is a summary of the purpose, methods, findings and conclusion of your project. It should be no more than 200 words. It should be clearly and concisely written. Provide only the most pertinent information, avoid citing references and include a brief statement of your main conclusions.*

# Introduction

This section should introduce the project. It should include an explanation of the problem and the objectives of the project.

The report as a whole should include a clear description of the lifecycle stages undertaken and must describe the use of appropriate tools to support the development process. It should give a full and accurate description of the work done and achievements made, together with complete software documentation and a user manual. Every effort should be made to provide a professional, quality description of the work. Proofread carefully for grammatical, spelling and punctuation errors or inconsistencies.

The report should be formatted as a justified, double-column, single-spaced, 10pt Times New Roman font document using an appropriate word processing system such as Microsoft Word, OpenOffice Writer or LaTeX and converted to a PDF file. The report should not exceed 15,000 words or 15-20 pages in length (excluding appendices). What is required is quality rather than quantity. The general layout of the report should follow this example document *although the number of sections and their headings will vary from project to project*. The report should be written in a formal style: it is neither a diary nor a magazine article. All pages should be numbered. All references should be cited in the main body of the report and a standard referencing format (such as IEEE or Harvard style) should be adopted [1, 2]. The report should demonstrate that the student has used appropriate tools to support the development process and that verification and validation have been applied at all stages [3].

The potato plant has played a central role in the world’s nutrition ever since it has been exported from the South American Andes and is the 4th most important food crop after Rice, Wheat and Maize. The potato plant tuber, the edible portion of the plant, is very perishable in comparison to other mass-nourishing products minimising export and trade with potato tubers. Regardless of the intention for potato farming, the potato is susceptible to a wide range of diseases which can have a dramatic impact on the yield of the farming and has had impacts on entire nations such as Ireland in 1740-1741. While the knowledge and use of pesticides and the knowledge about diseases have increased ever since the famine, it is still crucial to educate farmers and provide easy identification tools, especially to farmers in developing countries.

This project aims at creating an initial diagnosis tool for potato diseases and pests for farmers in developing countries who have no easy access to Microbiological diagnosis tools, the internet, pesticides or professional on-site help.

# Background

This second section would normally include a review of relevant literature and any similar products. The project should be placed in a wider context and this could include the scientific, technical, commercial, social and ethical context.

Current analytical tools for disease and disorder identification are a LFD field test, a FTA card test and two smartphone applications for site testing. The LFD test works similar to a pregnancy test in that it provides a quick way to test a leaf sample for a single trait of a disease. The drawbacks of the LFD test are, that a single test only tests for a single trait and there are hundreds of potentially devastating diseases. The FTA card test consists of a cardboard envelope onto which the farmer presses up to four samples of leaves. The cardboard is treated to conserve the genetic matter up to a decade, allowing the card to be sent to an institute with an appropriate DNA tester. The farmer would then receive an information package about the disease or pest with guidance on how to handle the problem. The benefits of the FTA card are, that the card is small and easy to ship worldwide. The DNA sample on the card can be thoroughly analysed to identify any traits of any known disease. The drawbacks are the duration of the test, as it could take weeks for the delivery of the card from remote areas, and extended times to analyse the sample.

A quicker alternative is a on-site evaluation of the plant, tuber or pest with a smartphone application. The International Plant Nutrition Institute (IPNI) offers an app which allows the identification of diseases in multiple crops [2], but only helps identify lacks of nutrients, and does not give suggestions to pests or diseases. The currently only alternative is "Potato Pests" by Leah Tsror, an Apple Application which lists and gives information about Diseases [1]. The benefits of this application are, that diseases are described accurately and symptoms are described. Drawbacks are, that the user is expected to know the name of the requested disease prior to using the application and that no advice is given to the user upon successful identification. The application is therefore not a useful first-diagnosis tool, but rather useful in combination with the FTA card, when the name of the disease is known and further education is requested.

More common techniques for evaluation are textbooks and field guides. While there are several editions in multiple languages, the range of examples is set, updates in research require buying a new edition and they hardly fit into trouser pockets, making them largely unavailable out in the field.

# Specification

You might choose to devote Section 3 to a specification of the problem and an explanation of how you arrived at this specification. An initial work schedule including an overall project plan with time-scales, deliverables and resources should be included in the report.

The problem identified is, that there is no quick way to analyse a potato disease or pest in the field quickly, as all the existing techniques require either external input or expert knowledge in the field. The product in design may therefore not rely on external input except for acquisition and updates, and may not rely on the assumption of expert knowledge in the field.

Initial Work Schedule

The timescale of the project is a 3 week period in which the entire research, planning, creation and evaluation cycle will be completed. As a agile approach was selected for the development of the project, production started early within the first week as a means of concept and feasibility testing. Key cornerstones of the project scope were the design of the database, the limitations of the XML protocol and the application implementation of the dynamic XML parsing GUI builder. The initial goal was to have the shell for the website, the database and the application done for the midpoint of the sprint duration to grant sufficient time for optimization, debugging and thorough testing.

# Design

You should include descriptions of the (user-centred) design methods employed to produce a usable product, including rapid prototyping, usability methods, results and re-designs as appropriate. Design decisions and trade-offs should be described, e.g. when selecting algorithms, data structures and implementation environments or when designing for usability. On certain occasions it might be helpful to refer to an equation, such as Equation (1).

*E = mc2* (1)

You will almost certainly want to use some figures or tables such as Figure 1 and Table 1. These should include captions.

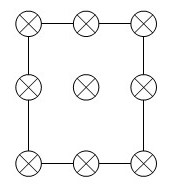


Figure 1. A diagram of a relevant pattern

Table 1. Performance of method A

|  |  |  |
| --- | --- | --- |
| **Test data** | **Error rate (%)** | **Time (sec.)** |
| *Set 1* | 70 | 3.1 |
| *Set 2* | 74 | 8.0 |

We designed the application on a cooperative design foundation, assuming that the user and we are on a similar level of experience when interacting with our products. The presumption was made on the basis that we, as computing students, know very little about potato plants that exceeds the store product, as would the farmer in Malawi, who may not be aware of the different kinds of diseases, but rather just aware of a ill plant. For the web interface, we assume that the data entry and editing will be performed by computing professionals with similar knowledge of databases as we have, meaning we can tolerate drawbacks in terms of design if it benefits the efficient manipulation of the database.

The first product created was a wireframe prototype of the application layout (Figure 1), which was revised roughly, but quickly agreed upon as the best design.

INSERT WIREFRAME

Figure 1. The Application Wireframe – Initial Design

We settled for the development of an Android application due to the availability of cheap android devices as oppose to highly expensive devices from Apple. It was discussed to provide a windows Phone implementation as well due to the involvement of Microsoft in Developing countries through offering windows surfaces and phones for promotional prices. The decision to not develop a Windows Phone application was made on the basis of time constraints and the desire to produce a reliable and thoroughly tested product rather than two buggy and untested products.

**Database Design**

We decided to have a database which holds the disease and symptom information as oppose to a text file or other data structure which would not allow the same extents of scalability as a database would. The database type chosen is a SQL type, as concurring database types such as Cassandra cater to large data amounts, and are only efficient if used on very large data sets, which this application will never reach, even if considering all possible future extensions.

The SQL database has two tables, one for the Diseases and pests and one for the symptoms. The database design allows for easy entry of new data and also uncomplicated altering of existing data. The most important quality of the database is easy querying of the data and conversion of the content to XML code, which can then be downloaded to the application from the application website.

**The website**

The website purpose is to enable the administrators of the application to add, edit and delete data from the database. The website is not intended for customer/user use and does not display data in an intuitive or readable format. This could be changed in later versions of the website, but is not currently implemented.

The current website features a login feature designed to protect the content from unauthorized access as well as ensuring integrity of the data through limiting the editing rights to trained and competent staff. The core of the website is a table showing the current database contents with the option to add a new entry to a table and edit or delete existing items. Once the changes are saved, the database will mark the changed content with current timestamps flagging them for update whenever a user checks for version updates to the current application. The design of the website is currently not aimed at a wide crowd of editors and contributors, as it is list based and does not provide an intuitive user interface. The customer statement to the matter is, that only one or two professionals will edit the application content, and therefore use the website, meaning that through training, these professionals can edit content correctly and likely more efficiently than through a complex user interface which would provide the same functionality.

**The Application**

The purpose of the application is to allow the user to navigate to a potato disorder or disease easily given only a couple of symptoms visible to an amateur. While the easiest option would have been to provide a list of all the diseases and have the user chose the relevant one, it would have been hard to handle with more than the first 20 diseases and therefore not scalable. We therefore settled on the idea of a decision tree which is modelled around symptoms. The user therefore starts out on the home screen of the application and is prompted with three categories such as "Pests", "leaf Symptoms" and "Tuber Symptoms". With the sample size of diseases reduced to one third, the user is prompted with the next layer of granularity such as "is the leaf crinkled or spotted". Each question reduces the available number of diseases and eventually guides the user to a narrow selection from which to choose. The greatest design challenge for this stage was to accommodate for multiple different symptoms which are all traits of the same disease, meaning that the database must accommodate for multiple symptoms for each disease, namely a linking table allowing multiple relations.

At the end of the decision tree, so when hitting the finest granularity which will leave only a couple of options to choose from, there will be a link to a detail page for a specific disease or pest. This detail page features all the data given to us from Prof. Lesley Doctorons with hyperlinks to further information. The information pages feature a design which aims to deliver data in the easiest possible fashion and avoids unnecessary confusion by first presenting descriptive and informative text for the disease followed by example pictures of symptoms which the user can compare to the example in hand.

Include wireframe designs

**The Update Function**

For the future of the app, it is vital that the app can be updated to keep up with current research and provide the best analytical options for a quick field diagnosis. For the purpose of updating the application contents, a timestamp has been added to all data in the database which is updated every time that data is created or changed. The user is given an update button on the home screen of the application through which the timestamp of the users’ last update is compared with the data in the database. Every item in the database which has a later timestamp than the users’ timestamp, is downloaded to the user application. The user application is then recompiled, creating the possibly new data structure of the application to incorporate the changes of the update. While the downloading of the files and the recompilation of the app structure take a lot of time, it is by far the best way to update the application, as downloading the entire structure every single time a change is made would take up much more time and resources and be potentially unnecessary, as changes might not change the layout of the decision tree at all.

# Implementation and Testing

You should describe important aspects of production, testing and debugging. Include a demonstration (or even a proof) that the specification has been satisfied.

Kurtis and Thomas were commissioned to design and develop the Database, Website and XML functionality while Stephanie and Ron shared the development of the Application, the XML parser, Ethics Applications and Project Management. The first version of the product was finished in the middle of the second week, so at the halftime of the project duration, leaving sufficient time to add data and test the functionality thoroughly. We found several minor bugs in the application during the beta testing phase such as invisible or hidden menus which were meant to be visible, text exceeding the previsioned space for the text and hyperlinks that weren't in the correct format. The database also turned out to update the timestamp too often, but this was resolved and should work in the final submission.

For our user testing, we were unable to contact actual potato farmers. We concluded, that we did not need or even want an educated and skilled potato farmer, as they were not necessarily the target group for the application. Given that the application is aimed at unskilled farmers who have no idea what problems their plant might have, we resorted to testing the application with subjects who had very limited knowledge about farming. Through this user testing, we found that the application concept is sound in general, as it allows the navigation to certain diseases from symptoms only. A suggestion we got from one test user was, that the granularity is often very exact, but also very tedious. So while there might only be one choice of a submenu within a menu, it still has to be opened to reach the disease page. This means that the user has to potentially navigate multiple symptom pages with only one option on each page, leading to a single disease. The solution we thought up for this problem is, to show all remaining results in a subsection below the menu options, to enable a direct skipping to the disease as oppose to forcing the user to navigate to the bottom of the menu. For time constraints, we did not implement this feature though, as it also meant to add an entry of each disease to each page of the parent symptom tree. The final product design would then be, that all diseases or all pests are shown when taping the "diseases" or "pests" button, and each sub question reduces the number of options left. This design may prove cluttered in the future if the application is scaled to several hundred diseases, but works fine for the moment.

Insert Screenshots of Application and Website

The specifications for the application were rather vague as the customer only had a vague idea of what the app could or should do. While the customer was certain about technique of search-by-symptom, other ideas were just proposed to the class as suggestions rather than mandatory requirements. Example suggestions which were deemed too complex for the project duration were optical recognition of pests and diseases or voice recognition for symptoms.

Keeping in mind, that the entire project duration including debugging and testing was only 3 weeks, the user specifications were followed and achieved to a high level of satisfaction. The application successfully allows navigation and diagnostics and will be scalable over a long period of time. The only notable drawbacks are the simple design of the web-interface for data manipulation, which could easily be reviewed though to provide a structure as it is found in the application to allow visualization for the administrators.

# Evaluation

## Usability

Usability should be evaluated.

## Other criteria

Other relevant criteria such as accuracy and computational efficiency should also be employed for evaluation as appropriate.

# Summary and Conclusions

Summarise the main points and ensure that you have described the final product. Include a critical appraisal of the project indicating the rationale for design and implementation decisions, lessons learnt during the course of the project and an evaluation (with the benefit of hindsight) of the final product and the process of its production (including a review of the plan and any deviations from it). Make recommendations for future work.

## Acknowledgments

The author would like to thank her wonderful supervisor, and her mum, dad, dog, cat and budgie for all their support.

The group would like to thank the James Huttons Institute for the Images and Information provided as well as the instant feedback received during the development process.

# References

[1] Tsror, Leah. *Potato Pests*. Computer software. *Leah Tsror, Ph.D. - ARO*. Vers. 1.1. Apple / ITunes, n.d. Web. 17 Aug. 2012.

[2] International Plant Nutrition Institute Apps. *Crop Nutrient Deficiency Photo Library App*. Computer software. *Crop Nutrient Deficiency Photo Library*. Vers. 1.1. International Plant Nutrition Institute, 9 July 2012. Web. 22 Sept. 2014.

# Appendices

The main body of the report should read as a self-contained document. However, appendices can be used for necessary supporting documentation. These should include a user manual, software, source code and minutes of your meetings *in electronic form only* (i.e. on a disc). Hardcopies of appendices should not be submitted.