

Pill Assisted Living (PAL)

Daniel Zinni
Nicholas Charlton
Brendan Shaklovitz
Camron Salisbury
Garrett Leach
Kevin Mott
Jonathon Kozak
Jia You (Justin)
Eric Hsu

Team URL: <http://utdallas.edu/~drz100020/>

Revision History

10/1/12 - Initial WRS

Process

Spiral Model

We are using a two phase Spiral Model for this project. The phases will each have all four activities of Elicitation, Negotiation, Specification, and Validation. This approach is iterative and will help modularize the process of developing an application for effective time and resource use.

AGILE

We are using the AGILE technique for this project. Since we are working with a smaller project, the low documentation requirements of AGILE are very convenient for us, and we can meet the requirement for high levels of communication by holding regular group meetings and by using email and text messaging to contact each other outside of group meetings.

Deliverables

9/8/12 - Preliminary Project Plan
10/2/12 - Phase I Interim presentation and Interim WRS document
10/18/12 - Phase I Final WRS document
11/13/12 - Phase II Interim Project submission
12/4/12 - 12/6/12 - Phase II Final Project submission, presentation, and demo

Team Members

Daniel Zinni
Nicholas Charlton
Brendan Shaklovitz
Camron Salisbury
Garrett Leach
Kevin Mott
Jonathon Kozak
Jia You (Justin)
Eric Hsu

Team Lead Schedule

There will be a different set of team leads for each phase of the project. Each team member must be a team lead at least once during the project.

PHASE I

a) Interim:
Brendan Shaklovitz
Nicholas Charlton
Daniel Zinni

b) Final:
Kevin Mott
Jia You

PHASE II

a) Interim:
Camron Salisbury
Jonathon Kozak

b) Final:
Garrett Leach
Eric Hsu

Meetings

Meeting 1

Date(s): 9/4/12 - 9/5/12

Location: GoogleDocs shared document

Agenda: Edit and complete the Preliminary Project Plan

Participants: Daniel Zinni, Brendan Shaklovitz, Eric Hsu, Garrett Leach, Camron Salisbury, Nicholas Charlton, Jia You (Justin), Jonathon Kozak, Kevin Mott

Summary:

Using email and text messaging to communicate with each other, we created, edited, and finalized the Preliminary Project Plan (PPP) through a shared document on GoogleDocs. All team members signed up to be team lead for a subphase of the project, and we were able to acquire a URL to host the softcopy of the PPP online.

Meeting 2

Date: 9/10/12

Location: McDermott Library, Austin Room 3.232

Agenda: Decide on an application idea and platform

Participants: Daniel Zinni, Brendan Shaklovitz, Eric Hsu, Garrett Leach, Camron Salisbury, Nicholas Charlton, Jia You (Justin), Jonathon Kozak, Kevin Mott

Summary:

We acquired a team URL, got everyone's student ID's, and discussed application ideas. The team chose the pill scheduler application idea on the Android platform. The team discussed details of how to communicate through comments in the GoogleDocs system.

Meeting 3

Date: 9/17/12

Location: McDermott Library, Austin Room 3.232

Agenda: Discuss the pill scheduler in more detail

Participants: Daniel Zinni, Brendan Shaklovitz, Eric Hsu, Garrett Leach, Camron Salisbury, Nicholas Charlton, Jia You (Justin), Jonathon Kozak, Kevin Mott

Summary:

We agreed on a team name, as well as discussed the goals, services, domain, constraints and mock up images of the pill scheduler. We began to outline the pill scheduler deliverable based on the sample given in the syllabus.

Meeting 4

Date: 9/24/12

Location: McDermott Library, Austin Room 3.232

Agenda: Clarify the presentation content and the responsibilities of the group members

Participants: Daniel Zinni, Brendan Shaklovitz, Eric Hsu, Garrett Leach, Camron Salisbury, Nicholas Charlton, Jia You (Justin), Jonathon Kozak, Kevin Mott

Summary

We decided on a presentation layout and divided responsibilities between team members

Powerpoint presentation responsibilities:

Title slide (Daniel)

Intro (Daniel)

Scenario (Nicholas)

What's needed to solve the problem (Camron)

Quotes/Research (Brendan)

Current Technology (Garrett)

Required Services (Kevin)

Required Features (Jonathon)

Development Challenges (Justin)

Conclusion (Eric)

References (All)

Meeting 5

Date: 10/1/12

Location: McDermott Library, Austin Room 3.232

Agenda: Make any edits to the presentation content and practice presenting

Participants: Brendan Shaklovitz, Eric Hsu, Garrett Leach, Camron Salisbury, Nicholas Charlton, Jia You (Justin), Jonathon Kozak, Kevin Mott

Summary

Made minor edits and practiced presenting our PAL presentation. Notified Daniel of what we covered in the meeting. Assigned any last edits to team members to complete before the presentation. Planned to create WRS document once the presentation had been completed.

Meeting 6

Date: 10/8/12

Location: McDermott Library, Austin Room 3.232

Agenda: Review our presentation as well as other team presentations, discuss things to add to the WRS document

Participants: Daniel Zinni, Brendan Shaklovitz, Garrett Leach, Camron Salisbury, Nicholas Charlton, Jia You (Justin), Jonathon Kozak, Kevin Mott

Summary

We discussed how our Interim presentation went, and analyzed the other team presentations to help decide how to expand the content of our Phase I Final deliverable. We added two new functional requirements to the WRS document, Security and Tutorial, that had existed in the design but not the specifications. We added details to our other functional and nonfunctional requirements. We also said we would add new screens to the mock-up to reflect modifications in the requirements. We also discussed creeping rates, why our project was the best one, and traceability. We agreed that what we talked about would be added to the WRS document. We stated that the new team leads for this subphase of the project were Kevin and Justin.

1. Introduction

We will develop an application to help schedule what pills (or vitamins) an elderly end-user needs to take and when. The application allows elderly people or their caretakers to quickly and efficiently set up their weekly regimen of pills without much room for user error. The application would allow users to choose names of common medications, the color of the pill, and the dosage of the medication. A user or caretaker could use a weekly calendar to pick what days they are supposed to take their pills, as well as a time for them to take their pills. When a user or caretaker has finished scheduling their medications, a colored dot representing the color of the pill will appear on the calendar slot in chronological order with any other pills that have been added for that day. The application would include a calendar for the user to look over whole week and add more pills to their schedule. When a pill was scheduled to be taken, the application would activate an alarm on the smartphone with the name of the pill they have to take, the dosage, and the color of the pill. If there are multiple pills in the same time slot they will all be listed.

2. Issues with Preliminary Definition Given

2.1 Domain, Stakeholder, Functional and Non-Functional Objectives - Issues

2.1.1 Domain feasibility

We need to test the overall accessibility of the system design in order to ensure that the end-user can use the system effectively and comfortably.

Options:

- Gathering opinions of our mock-up from end-users.
- Developing a working prototype and having end-users try to navigate the system and give their opinion.

The most feasible option is to gather opinions from end-users based on our preliminary mock-up. This option reduces the unnecessary effort and complexity of developing a functioning system in order to gather opinions

2.2 Software System Functional Requirements - Issues

2.2.1 Pill Alarm Images

When there are multiple pill images on an alarm, we need an efficient way of representing the pills without overcrowding the interface

Options:

- Use scrollable pill images and allow the end-user to confirm each pill individually.
- Use a single pill image which must be confirmed before the next pill image is shown.

The most feasible option is to use scrollable pill images with individual confirmations. This clearly shows the user that there are multiple pills scheduled for the time of day that the alarm is set off at, reducing ambiguity related to how many pills must be taken and confirmed.

2.2.2 Pill lookup

There should be a way for someone to look up a pill that may not have a bottle to identify it.

Options:

- A pill lookup feature will link with an online pill database where a user could search for pill colors, codes, and shapes.
- A large pill reference listing could be included in the default pill database including possibly outdated pill information.

The most feasible thing to do is to link to an online pill database through the application. This reduces development time in looking up and adding a large amount of pills for reference, and reduces the overall size of the application.

2.3 Software System Non-Functional Requirements - Issues

2.3.1 Interface

The interface needs to be easy to read and understand. A user should not have to go through a complicated process to learn how to use the system.

Options:

- Large text and a simple interface.
- A “simple” version of the application and a “complex” version of the application to support different users of different skill levels.

Using large text and a simple interface is the more feasible option in order to ensure readability and an easy understanding of the system. It would significantly increase development time if two separate versions of the applications had to be implemented.

2.3.2 Separation of Environment

There needs to be a separate environment for a user entering pill information, and a user interacting with alarms and viewing the weekly calendar.

Options:

- Develop separate applications for those interacting with alarms and those interacting with the scheduler.
- Include an “Advanced” section for users to enter pill information, and have the weekly pill calendar on a separate screen. Allow users to “lock” the advanced pill scheduling features to prevent accidental changes to the schedule.

It is more feasible to “lock” advanced features like pill scheduling once the schedule has been entered instead of developing separate applications. This reduces overhead and development time when we start to implement the system.

3. WRS

3.1 W - World

3.1.1 Problem

It is important for those needing daily medication to take it regularly and correctly, but this doesn't always happen. Those taking the medication may have trouble remembering to take the pills they need, or it can be complicated for them to manage multiple medications. Because medications may be missed or forgotten, the health of those taking the pills can be at risk.

3.1.2 Goal

We will develop an application aimed at reducing the complexity of remembering and maintaining a pill schedule. Once set, a pill schedule in our system will keep track of a wealth of information related to the medications and only require the end-user to confirm that they took the pills entered in the schedule.

3.1.3 Improved Understanding

For our system to work, we have to collaborate with elderly end users in order to ensure that our system is easy to use and effective in maintaining a complex pill schedule. We will have end-users evaluate our mock-up of our proposed system in order to decide if any changes need to be made to the application interface.

3.2 RS - Requirements Specifications

3.2.1 Functional Requirements

The functional requirements for this project must allow end-users to manage multiple reminders to take pills. Supported features include:

- Customizable schedule for multiple kinds of pills (or vitamins)
 - Input: When a pill needs to be taken, and what kind of pill needs to be taken.
 - Output: A weekly view of the pills to be taken for the week is displayed on the home screen.

A user will be able to enter multiple pills into the schedule to be taken at multiple times during the day or week.

- Alarms that let end-users know when to take the medication
 - Input: Information from the customizable schedule.
 - Output: An alarm is activated at the time a pill needs to be taken.

An alarm should be activated whenever a pill has been scheduled to be taken. The alarm should include the information entered into the pill schedule.

- Specific pill color and dosage information
 - Input: Pill color and dosage information is added to the associated pill.
 - Output: Pill color and dosage is displayed on the alarm associated with the pill.

There will be options to add specific pill information, like color and dosage, to the schedule that will be shown whenever an alarm associated with a pill is activated.

- Instructions on what to take pills with
 - Input: Instructions about how to take the pills correctly (with food, water, etc) are added to the associated pill.
 - Output: Instructions are displayed when the alarm associated with a pill is activated.

There will be options to add specific instructions on how to take a certain pill to the schedule that will be shown whenever an alarm associated with a pill is activated.

- Feedback

When a pill has been consumed for a caretaker:

- Input: A user confirms that they took the pill using the system interface twice, once for initial confirmation on the alarm screen, once for final confirmation on a subsequent screen.
- Output: A paired app notification, text message notification, email, or automated phone call is activated to let an outside entity know a pill has been taken and not missed.

When an alarm is activated and a user confirms twice that they took the pill associated with the alarm, a notification will be sent (via a paired app notification, text message, email, or automated phone call) confirming that the user took the associated pill.

When a user can't find the required pill:

- Input: A user chooses the option on the alarm screen that they "can't find the pill" twice, once for initial confirmation on the alarm screen, once for final confirmation on a subsequent screen.
- Output: A paired app notification, text message notification, email, or automated phone call is activated to let an outside entity know a pill was not taken because the user couldn't find it.

When an alarm is activated and a user chooses the option on the alarm screen that they "can't find the pill" and they confirm this selection, a notification will be sent (via a paired app notification, text message, email, or automated phone call) to let an outside entity know the pill was not taken because the user couldn't find it.

When a user ran out of the required pill:

- Input: A user chooses the option on the alarm screen that they "ran out of the pill" twice, once for initial confirmation on the alarm screen, once for final confirmation on a subsequent screen.
- Output: A paired app notification, text message notification, email, or automated phone call is activated to let an outside entity know a pill was not taken because the user ran out of it. An automatic refill request is made to an associated pharmacy.

When an alarm is activated and a user chooses the option on the alarm screen that they "ran out of the pill" and they confirm this selection, a notification will be sent (via a paired app notification, text message, email, or automated phone call) to let an outside entity know the pill was not taken because the user couldn't find it. If the application is set up with prescription refill information and a pharmacy location, the prescription refill request is processed automatically for pickup at the pharmacy via an available internet connection.

If internet access is limited, the application can queue notifications that require internet connectivity, such as sending an email or notifying a paired app. If the user is interacting with the application through a tablet without a dedicated data connection, the user must eventually gain internet access to send pill notifications.

- Pill lookup for pills without bottles

- Input: Information about an unknown pill (color, shape, numbers).
- Output: Online search results that most closely match the unknown pill information.

When a pill doesn't have a bottle, or information about it is missing, a user will be able to look up a pill based on its color, shape, and any numbers on the pill. The application will link to an online pill database so users can easily identify outdated or unknown medication. An internet connection is required to use the pill lookup

feature.

- Security features

Backup functionality:

- Input: A user clicks the “Backup” button in the “Security Options” section of the application.
- Output: A file containing the user’s application data is saved locally to the phone and may be exported.

When the user wants to save their pill and schedule information, they may backup their data to a local file that is stored on the phone and may be exported to an external server or device for safekeeping and restorative purposes.

Locking the application:

- Input: The user clicks the “Set a Password” button in the “Security Options” section of the application.
- Output: The application requests a PIN or password to lock the application’s “Advanced” section and prevent tampering.

When the user or caretaker wants to prevent accidental or intentional changes to the pill schedule or other advanced features, they may choose to click the “Set a Password” button and set a PIN or password to prevent changes to the application. A locked application still allows viewing of the pill schedule and tutorial as well as activation of alarms, but prevents users from accessing “Advanced” features without entering a correct PIN or password. For example, a user may click “Advanced” to see the buttons, but when they click on one of the buttons a PIN or password prompt appears, and a correct PIN or password must be entered to continue. A user can turn off the lock or modify the locking mechanism with the “Locking options” button.

- Tutorial

- Input: A user clicks the “Information” button on the home screen.
- Output: Text and simple pictures describe correct use of the application.

When a user wants more information about the application, they can click the “Information” button on the home screen to get a simple and clear guide on how to use the application’s features.

3.2.2 Non-Functional Requirements

Compatibility

- The system we are developing needs to be compatible with multiple device types.
 - The application will be compatible with both smartphones and tablets.

Availability

- Our system needs to be widely available and easily accessible.
 - We will develop it on the Android platform and release it in the Google Play store.
 - An internet connection is not required to use the main features of the application like pill scheduling and alarms, so the application is available without a live connection. However, an internet connection is required for features like paired app synchronization, pill lookup, and email notifications.

Affordability

- The application we develop must be affordable in order to prevent people who could greatly benefit from the application from refusing to use it based on how expensive it is.
 - We will moderately price the application.

Usability

- The system interface must be easy to understand and use, and must require a minimum learning curve for a user to navigate the application.
 - All essential application services (view schedule, alarm and confirmations, tutorial) are 0-2 clicks away from the main screen.
 - Advanced services (set schedule, lock application, pill lookup) are 1-3 clicks away from the main screen.

Aesthetics

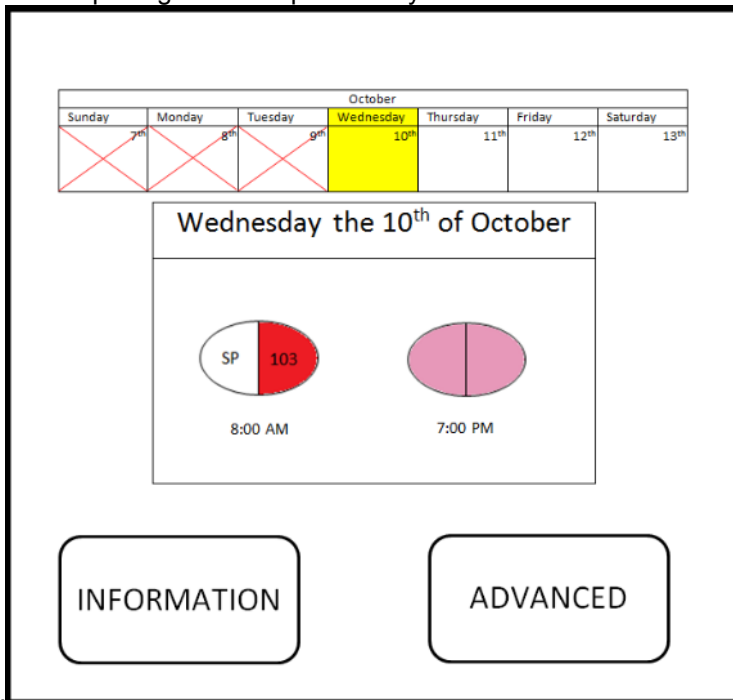
- The system interface must be simple and easy to read for all users.
 - Large fonts and buttons will be used.

Separate Environments

- The advanced pill scheduling part of the application must be separate from the weekly calendar view and alarm functions of the application.
 - We will include a “locking” feature which will prevent accidental changes to the schedule once it is entered.

4. Preliminary Prototype and User Manual

Below are mockup images for the preliminary



prototype.

Figure 1: The home screen of the application.

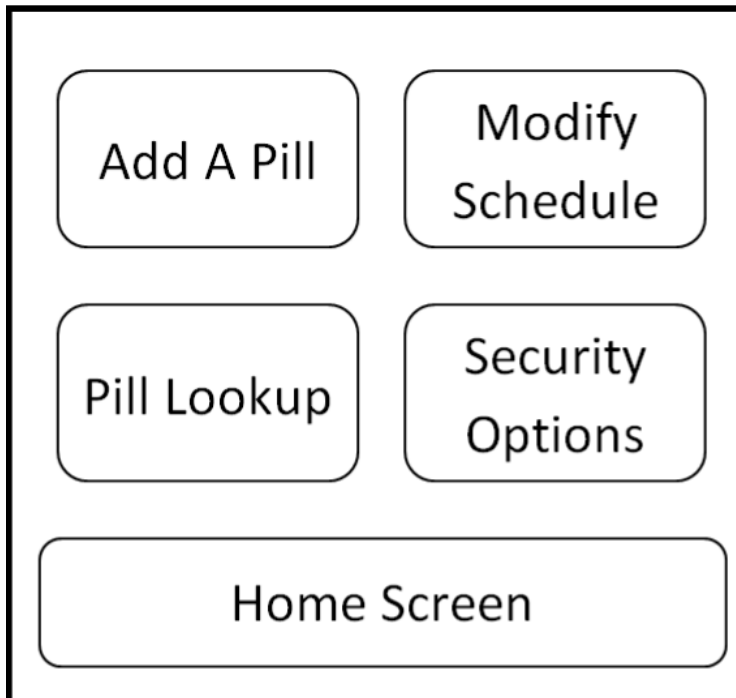


Figure 2: The “Advanced” screen of the application, reached through the home screen.

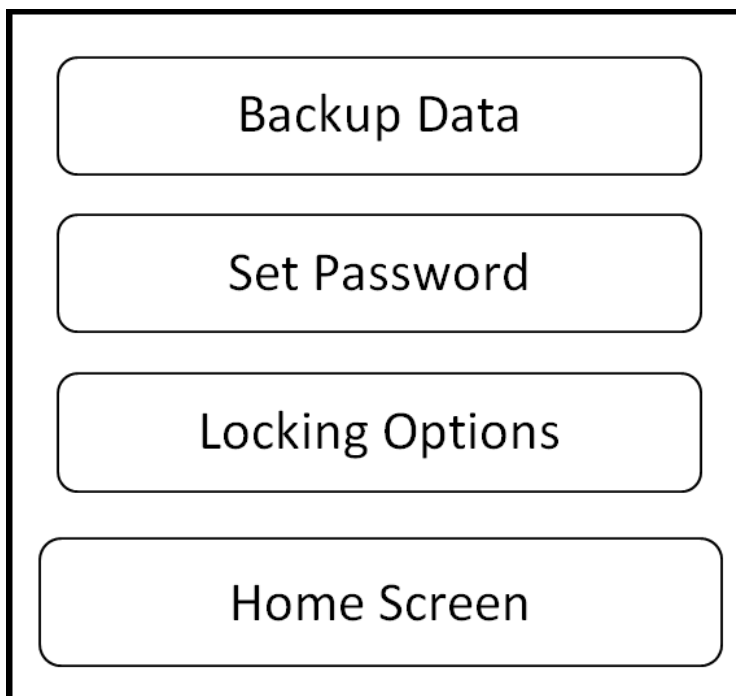


Figure 3: The “Security Options” screen of the application, reached through the Advanced screen.

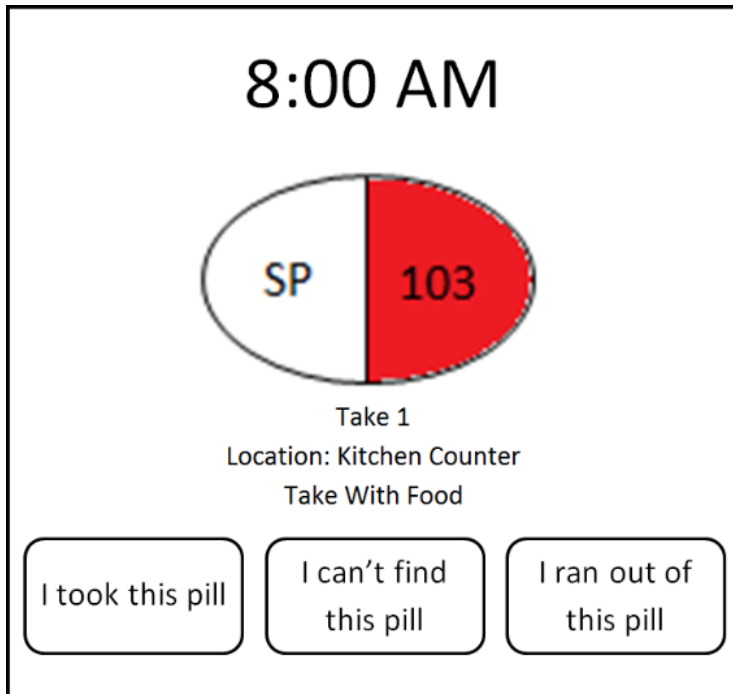


Figure 4: The alarm screen of the application, activated by the pill schedule when a pill must be taken.



Figure 5: The confirmation screen of the application, when the user selects an option from the alarm screen.

5. Traceability

For our application, our goal is to at least guarantee forward traceability, but to aim for backward traceability as well. For this phase of the project, the traceability is between our functional requirements and our design prototype. For forward traceability, we made sure that every functional requirement corresponded to an element in the design of the prototype:

- Custom schedule - Under the “Advanced” section, users can modify a pill schedule. The weekly pill schedule is displayed on the home screen. The pills to take for the day are also displayed on the home screen, along with the times to take the pills.
- Alarms - An alarm is activated when the current time corresponds with a time set in the schedule when a pill must be taken.
- Pill color and dosage information - The color and dosage of the pill are displayed when the alarm is activated.
- Instructions on how to take a pill - Instructions on how to take the pill are displayed when the alarm is activated.
- Feedback - When an alarm goes off, three buttons on the alarm screen provide three different feedback functions for a pill: the user has taken the pill, the user can't find the pill, the user ran out of the pill. When any of these buttons is pressed and confirmed, a feedback notification is sent to an external entity.
- Pill lookup - Under the “Advanced” section, users can look up pills that they are unfamiliar with.
- Security - Under the “Advanced” section, users can choose to backup their data to their device, set a security password, and modify locking options.
- Tutorial - On the home screen, a user can select the “Information” button to view an easy to understand tutorial about how to use the application.

6. Application Superiority

For our application, we focus on a specific user group: the elderly. We concentrate on usability and functionality issues that directly affect this user group, and do not sacrifice quality of the application for a larger scope. Some groups implementing similar features to ours have stated that they aim their application at a large range of users, including vastly different ages and different comfort levels with technology. We feel that even if they were to implement all of the same features that we have, they would suffer from a deterioration in the overall quality of the application because of their wide scope. Our team believes that large differences in the users an application is aimed at will result in many people not getting what they want or need out of the application. Elderly users may need larger text and buttons in order for them to be comfortable using an application. Those who are not that technologically savvy will feel alienated by an application that is too complex for them. On the other hand younger users may want more content on the screen at once, and those comfortable with technology will crave more complexity in the application. Teams that make their scope of users too large are unlikely to please everyone. Since our application is focused on a specific user group, we feel that we will be able to make the majority of our users very satisfied with our application. We can efficiently design an application to accommodate our user group very well, instead of trying to cater to interests that may be vastly different.

Similarly, some teams decided to design an application with a suite of features in different categories of aid. While they may be able to implement all of their planned features, they will likely not have the time or resources to implement all of the features with a high amount of quality. It is more plausible that they will do many things OK, instead of doing one thing very well. We believe that by focusing our attention on an application to solve a specific problem we will be able to efficiently alleviate health risks related to pill management.

Finally, we chose to address the problem of pill management because of its health implications. There are many people who have a complex, confusing pill schedule. And they may not always be able to execute their schedule correctly due to time constraints, lack of organization, or a failing memory. When someone doesn't take a pill that they need in order to maintain good health, they are putting themselves at risk of harm or death. We felt that the health of the elderly was the most important issue for us to tackle with our application.

References

<http://www.utdallas.edu/~chung/CS4351/syllabus.htm>

Appendix

Compatibility: The system we are developing needs to be compatible with multiple device types.

Availability: Our system needs to be widely available and easily accessible.

Affordability: The application we develop must be affordable in order to prevent people who could greatly benefit from the application from refusing to use it based on how expensive it is.

Usability: The system interface must be easy to understand and use, and must require a minimum learning curve for a user to navigate the application.

Aesthetics: The system interface must be simple and easy to read for all users.

Separate Environments: The advanced pill scheduling part of the application must be separate from the weekly calendar view and alarm functions of the application.

Index

Creeping Rates

We anticipate a creeping rate of 62.5%, because we are prepared to accommodate changes to five out of eight of our functional requirements. We decided this by dividing our functional requirements into two categories: critical and noncritical. Critical application features are required in order for the application to function correctly. Noncritical application features enhance the application's scope and usefulness, but are not required for functionality.

The critical application features include:

- Custom pill schedule: Needed to manage multiple medications.
- Pill alarm: Needed to remind the user about multiple medications.
- Pill color and dosage information: Needed to inform the user about how to identify multiple medications.

These features are required for the application to function.

The non-critical application features include:

- Instructions on how to take the pills: Redundant information that is also on the pill bottles.
- Feedback services: Additional information provided to an outside entity, but not a required function for the critical features of managing and maintaining a pill schedule.
- Pill lookup: Useful for identifying unknown pills, but most pills will already have a bottle, and pill lookup does not directly relate to the critical features.
- Security: Prevents accidental or intentional changes to a set schedule, but not a requirement to satisfy the critical features.
- Tutorial: Helps users find their way around the application, but not needed for the critical features to work.

These features enhance the application, but are not absolutely necessary for functionality.