ABSTRACT
The MoodTracker project goal is to create a technological aid for therapy patients and those implementing cognitive behavior therapy. This paper details the process of design, testing, and refinement of the MoodTracker application. The application is written for android mobile devices. Demonstration Video

Keywords
Cognitive behavior therapy, technology, mobile application, user interface

1. INTRODUCTION
Cognitive Behavior Therapy (CBT) is a form of talk therapy that uses a systematic approach to address mental issues such as depression, eating disorders, or mental illness. A major component to CBT is tracking a patient’s thought patterns to treat their disorders (Martin). CBT is most effective when patients continue their work consistently between therapy sessions, however, it is hard to remember to record and track moods over time. Even with consistent journal work that recorded data is not easily usable. It would take serious effort to find trends within written journal entries.

MoodTracker is designed to address these issues involving the application of CBT. The goals of the project are to provide a convenient, fast, easy way to record mood information, then process this information to make it more useful. The end goal is to help users get the most out of their therapy process.

2. CREDITS
Troy Crosby: Preliminary design, Design Iteration, Prototype development, Usability Testing
Brian Lager: Preliminary design, Usability Testing, Compiled usability test packets
Matt Starsoneck: Preliminary Design, Prototype Development and Improvement, Usability Testing, Tutorial Video
Paul Hazlet: Prototype Development, Design Iteration, Editing, Usability Testing
Tobin Valenstein: Original idea and design, Preliminary Design, Design Iteration, Usability Testing, Prototype Development and Improvement, Editing, Team Lead

3. PREVIOUS WORKS
eMoods Bipolar Mood Tracker is a mood tracker designed specifically for those who are bipolar. Like MoodTracker it offers the ability to track medications taken daily. It has four variables which one can input intensity levels for, these variables being depression, anxiety, elevated, and irritability. It also offers a graph displaying the different data points, but shows four lines, each representing a variable. One cannot say they are happy or content in the app, which means it's very negatively focused.

Mood Panda is a mood tracking app developed for the iPhone. It offers several features similar to those proposed in MoodTracker such as rating mood as on a number scale, providing visualizations such as
graphs of mood versus time, and journaling capabilities. However, MoodPanda is intended to be used as a social application, which is a deterrent to certain users, not everyone is comfortable sharing personal information like emotions to a wide audience.

*Mood 24/7* is a mood tracking website, but uses text messaging to keep track of moods rather than having a mobile application. This limits the portable features of the service and making edits or changes are not possible to do through SMS messaging. It does allow health professionals to follow a patient's mood timeline.

*T2 Mood Tracker* is a mobile application developed by the National Center for Telehealth and Technology to improve the health of military personnel and veterans. The application comes preloaded with anxiety, stress, depression, brain injury, post-traumatic stress, general well-being categories each with ten or more variables that can be set using sliders. The ability to create custom categories is available. It also has a note taking feature.

4. DESIGN

4.1 Design Process

The MoodTracker application was conceived as an application that would provide a tool for therapy patients to track their mood over time. This was identified as the main feature, with the goal of providing valuable data to patients and therapists that would improve the therapy process. After team discussion and research, we also identified the needs to be able to express the data gathered in a graphical manner, and to be able to track the duration a medication is taken. These user needs seemed the most important. After identification of these user needs, and compiling a list of core features, our group decided to create multiple prototype designs of the application interface. The next step was to combine all of the best ideas from the initial designs, including what widgets were to be used, what would be displayed on each tab, etc.

After compilation of our separate designs into a rough group-approved design, we began to complete a programmed interface using the Android SDK.

The first step in the compilation process was to consolidate the four low-fidelity prototypes that our team created into one design. Since each low-fidelity prototype was designed individually, many of them implemented features in the same way. Some of these common features were separating the functionality onto 3 main screens, having the main screens be accessible via tabs at the top of each screen, and using a slider to input the user’s mood score among others. Features that were common across the majority of the low-fidelity prototypes were chosen to be included in the high-fidelity prototype as well. For the remaining features that were implemented in different ways our team discussed the merits of each design before choosing which implementation we would include in the high-fidelity prototype.

Our interface was to have an overall structure of three tabs that separated the MoodTracker application into three logical sections: Mood Input, Graphical Representation, and Medication Tracking. These tabs allow quick access to all parts of the MoodTracker application to the user. This is extremely important in for our mobile application, since the faster and easier it is to use, the more likely the user will consistently use it. Consistent usage is vital to effective mood tracking.
4.2 Design Choices

The Mood Input (fig. 1) screen is the first screen that the user sees when opening our application, as the user is expected to use this function more than anything else on the app. On all of our screens, we decided to compress what the user has to input into a resolution that would prevent unnecessary scrolling. Therefore, our group chose to utilize widgets that popped-up secondary input menus rather than clutter up our screen with multiple calendars and time inputs. This priority allowed us to keep the simplistic ideal that we strived for in our initial design, while still providing the same amount of functionality to our user. We decided to have two ways of recording mood, a numerical slider value “How are you feeling?” and a descriptive mood in a drop down menu. This would allow the user to track their mood in two ways, intensity and adjective. By using this method, we could cater to all kinds of therapy patients, including those who have extreme mood swings (intensity) and those who simply need to describe their feelings in a concise manner (adjective). Finally, we decided to include an optional note in case the user wanted to make a note of what happened that made them feel this way, considering what happens in a person’s social and professional life can have a clear effect on their mood at the time.

The Graph Mood Screen (fig. 2) provides valuable feedback to the user, which is an important reason to use our application. Simply tracking the mood is useless if the data can’t be compiled and easily relayed back to the user. Our group decided that multiple graphs were the way to go, involving a line graph that would track mood intensity, and a circle graph that would track percentage of different moods felt. The design is simplistic enough, utilizing two drop down menus to tell the program exactly what time and medication the user would like to see displayed on the graph. After this input is complete, the user has the choice of four buttons: Line Graph, Circle Graph, Clear Graph and Export Data. Clicking Line Graph or Circle Graph will utilize the user input and stored mood data to generate a graphical representation to display to the user. These graphs appear directly over the Time Period to be displayed and Medication to be displayed drop down menus to maintain our design concept of concise screens that stray away from scrolling. The graph can be easily cleared with one click of the “Clear Graph” button. The “Export Data” button prompts the user for an e-mail address that they can send the data to – perhaps a therapist who would want to use this data to better assess their patient’s state of mind.
The Calendar/Meds Screen (fig. 3) provides the user with an easy way to view their mood on a certain day, and to view and edit the medications that they currently are on. The calendar allows for the selection of dates, which allows prevents the user from unnecessary text input, and simply allows them to utilize the best part of the mobile device: touch input. This allows them to quickly pull up information on how they felt on a particular day, which is important data, especially for a therapist to see. The user can click also on View/Edit Medications or View/Edit Alarms, which takes them to a separate screen. Although this is slightly different than the other two screens, in this case, it is extremely important to have a secondary screen that can display a large amount of text data, as it is not known just how many medications that the user would like to keep track of using the application. Perhaps they would just track their prescribed medications, such as Zoloft, but leaving the option open to track Vitamin supplements or other health medications allows for the most flexibility and appeals to a larger user base.

4.3 Transition Diagram
5. DEVELOPMENT PROCESS

The first step in the development of MoodTracker was to decide, based upon our problem, design approach and target audience, on a set of core features that would make this app both practical and useful to a user. We began this process generating several user scenarios. This method allowed us to quickly elicit a set of core features from a project idea that up to this point was not particularly well-defined. In cases such as this, Shneiderman states that “many designers have found day-in-the-life scenarios helpful to characterize what happens when users perform typical tasks” [2]. Each of these scenarios describes a typical use of the MoodTracker app by a specific user, such as a middle-aged male using MoodTracker to help with a current therapy program or a young female who is using MoodTracker to complement self-help books she is reading. By analyzing these detailed scenarios we were better able to understand what features would be important to our users in various situations. Here is one example of a user scenario that our team created.

4.1 Feature List

For each of these scenarios a target user is proposed along with their reason for using MoodTracker. As the scenario develops the each user’s specific needs are addressed by suggesting possible features that would allow them to accomplish their goals. Using several user scenarios and further brainstorming within our team a list of core features was compiled. The features describe the core functionality of the application, and how they would work together. The following is the list of core features that were decided to be critical to the MoodTracker application.

- Input mood data, such as, emotion, mood intensity and related notes
- Input medication data such as type, dosage and duration of use
- View/edit previous mood and medication entries
- Visualize data in chart/graph form
- Export data via email to a personal computer in excel format
- Set alarms/notifications to prompt user input

4.2 High Fidelity Prototype

The next phase in the development of the MoodTracker app was to create a high-fidelity prototype that would be used for usability testing.

The first step in this process was taking the low fidelity prototypes, combining them and improving on their design. Some features were re-organized to be on different screens, or combined to make mash-ups of implementations.

After determining an overall design we implemented our high-fidelity prototype as an actual Android application that ran on an Android emulator. This was important for the success of the usability study. One of the guidelines for usability testing presented by Dr. Shneiderman is to “Create prototypes using the most appropriate technology for the phase of design, the required fidelity of the prototype and the skill of the person creating the prototype” [1]. The strength of evidence related to this guideline was rated as a 3 out of 5 which we believed would make it a solid factor in the future development of MoodTracker.

After completing the implementation of our high-fidelity prototype we produced screen shots of each screen in the application which are in the following transition diagram.
4.3 Usability Testing

In order to assess the effectiveness of our high-fidelity prototype we created a usability test to administer to potential users. The test consisted of four sections, the test instructions, a pre-test questionnaire, a series of tasks for the test subjects to complete and a post-test survey.

4.3.1 Test Introduction

The introduction to the usability test serves two main purposes. First, it explains to the users the overall purpose of the testing and assures them that it is the interface being tested and not their abilities. Second, it goes into detail about the structure of the test. This includes details explanations and instructions for the following three portions of the test. The introduction was designed to be short but properly prepare the testers for the usability study.

4.3.2 Pre-Test Questionnaire

The pre-test questionnaire consisted of four simple questions that helped us to categorize our users in regards to our target audience. They also provided helpful information for the analysis of the post-test survey questions.

4.3.3 Usability Tasks

The usability test consisted of a series of tasks the tester would carry out. We gave each tester the same four tasks. Two of these tasks were simple tasks which had the user complete an mood input entry that was contained within one screen of the interface. The final two tasks were more advanced tasks, which integrated the simple tasks with navigating to new screens within the application and using different features, such as the add a medication functionality. The list of tasks was designed to have the users navigate through the majority of the interface, and interact with the most important screens and features.

Figure 5: High Fidelity Prototype Transition Diagram
With this experience the users were able to easily and accurately answer the questions that we posed in the post-test survey.

4.3.4 Post-Test Survey

The post-test survey consisted of six questions. The first 3 questions were semantically anchored survey questions that asked the users to rate the ease of using each of the 3 main screens on a scale between 0 (easy) and 9 (difficult). Using these questions gave us access to numerical data that allowed us to easily graph and spot trends in the user’s responses. The following chart summarizes the data gathered from each of our 5 users.

![Post-Test Survey](image)

This chart clearly shows that overall the users had the easiest time understanding and using the mood entry screen, followed by the graph screen and they had the most difficult time with the medication entry screen.

The final 3 questions on the post-test survey were short answer questions, asking the user to share what they felt were the best and worst part of the interface as well as any additional comments that were not covered in previous questions. The answers that we got in these questions were strongly related to the user’s opinions on the first three questions. However, these questions offered greater insight into what specifically the users liked and what was causing them difficulty. We used these answers to create short summaries of the experience each user had. Here is an example of one of these summaries.

4.3.5 Usability Test Summary Example

56 year old, male, computer engineer, no Android experience

This test subject was an older user who was competent with computers and modern technology but has had no experience with Android devices or other smart phones. Overall he had no trouble in completing the usability tasks, despite his lack of experience with Android devices noticeably slowing his progress. His favorite part of the interface was the mood input screen. He liked the fact that with the exception of the user note field all of the input fields had a restricted number of choices. Being a novice user he really appreciated how this guided his input and prevented him from making errors. In regards to the rest of the interface he felt that several aspects seemed out of place. One such observation was that the alarms screen was accessible through the calendar/meds screen which was not at all obvious. Similarly he felt it was not obvious that the export option should be under the graphs screen. Overall he thought that the app could be better organized but admitted he would not be comfortable with many more screens. Another aspect that disturbed him was that the time entry field for the alarms screen did not function in a similar way to the time entry screen on the mood input screen. Finally he was troubled by the lack of guidance in input on the medications screen and thought it might be missing fields such as the frequency that medication was taken.

6. REVISIONS

From all of the data that we gathered in our usability testing we created a list of revisions that we thought were important to the final design of the MoodTracker app. The following is a summary of this list of revisions.

6.1 Final Status

The final status of our project is that of a mostly complete user interface, but without any backend functionality implemented. Additionally, there exists some room for improvement in the user interface. Currently there are no tabs in the interface, despite them being part of the design. This was a development decision made based on current time constraints. There are also some minor improvements that can be made by employing the use of custom graphics.
Our application employs a three-tab interface which represent the three main functionalities of the application. The first (default) tab (fig. 7), for mood input, allows users to record their mood both quantitatively and qualitatively. Quantitatively, a slider is used to record a user’s basic outlook in comparison to two extremes represented as a frowning face and a smiling face. Qualitatively, their mood can be recorded through a spinner populated with a list of different emotions. An optional note field exists where users can log details of what caused them to feel the way they currently feel. A submit button is used to record this date. At the current time a backend database has not been implemented to store the information that is inputted through the interface. If the user wishes to record information for a data and time other than the current data and time, two buttons exist at the top of the screen which open date and time selectors. The chosen date/time are displayed next to these buttons, and by default show the current date and time.

The second tab, labeled “Graph Mood” (fig. 8), allows the users to view their mood history as a pie chart, a line graph, or via a comprehensive log that can be emailed to them. Through two dropdown menus users can choose a period of time for which to display data, or the span of a specific medication for which to view data. Buttons prompt the display of a pie chart, to show the breakdown of qualitative mood data over time, or a line graph, to show the quantitative progression of mood/outlook over time. Additionally a button exists to clear/return from the graph, and a button exists which prompts the user for an email address to which a comprehensive log of mood data will be sent. Currently the data displayed through this tab are all imaginary placeholders, as we do know have real data stored in a backend database.
The third tab, or Calendar tab (fig. 9), is where users can perform all actions that are related to specific dates or times. The content of this tab consists of a calendar widget, on which dates could be clicked to reveal a popup of mood information recorded on that date, along with medications they are on. Two buttons also exist on this tab, each bringing users to a new screen, which allow you to view/add/edit medications and view/add/edit alarms. Users can input medication names and dosages, although these medications are not currently tagged to a time period. These medications are shown as a list which users can click to edit. The alarms button leads to a similar list of alarms users have created and can edit. Users can add a new alarm by specifying a time and a frequency of recurrence. Both medications and alarms are currently not stored in a backend database, meaning the calendar is unable to display actual information for specific dates.

6.2 Future Development Plans

Work that still needs to be completed before the application is fully functional can be described as five distinct tasks. First, the tab structure of the application needs to be implemented again to function properly, as it stands now the application uses buttons instead of tabs. Second, a backend database must be implemented into which the mood input and medication/alarm data can be stored. Having this data retained is necessary for the implementation of the remaining three tasks. Third, graphing functionality needs to be implemented drawing on the mood data stored in the database. Android libraries already exist that can accomplish this task, but it is dependent on a backend database. Finally, email functionality must be implemented to send a comprehensive mood/medication log to users, and the alarms/reminders set in the calendar tab need to make system-level calls to set/edit the device’s built in alarms.

6.3 Recommendations to Future Developers

Two major recommendations we would make to future developers both revolve around the graphing portion of the application. First we would recommend that future developers carefully choose what graphing library or classes they use within the application. Extensive, non-native libraries may be less reliable, slow, and subject to change compared to other solutions which may be part of Android’s native codebase. Additionally, there is design choice in how the graphs are displayed. Graphs could be displayed in a new screen, at full-screen dimensions, or remain as a popup. However the graphs are implemented, we would recommend additional usability testing to assess the ease-of-use or ease-of-comprehension for the graphs. This was a factor we were only able to test to a limited degree due to the use of placeholder graphs and lack of real data.

7. ACKNOWLEDGMENTS

Our thanks to participants in our usability study and to Dr. Schniderman for his input and guidance.

8. REFERENCES


[3] Morfeld, Matthias, Corinna Petersen, Anja Kruger-Bodeker,


