

Alexis Barltrop Meiqi Sun Percy Hu Tyler Baker

Preface

The following report is the original work of the following students as part of the COSC345: Software Engineering paper at the University of Otago.

Alexis Barltrop	ID: 7176303		
Meiqi Sun	ID: 5926587		
Percy Hu	ID: 2856476		
Tyler Baker	ID: 1927791		

Executive Summary

In this report we comment on the usability and performance of the first version of our habit manager application. The iPhone application was tested by members of the development team and other external testers. Both physical iPhones and the Xcode iPhone simulator were used in testing.

User feedback suggested the app was easy to use, visually appealing and useful for managing a number of different habits or schedules. Users also provided feedback on features that would improve the application including an edit function, colour coding of different habit types and the option to make a monthly habit. This report also contains a schedule for the implementation of these new features, aiming for the final version of the software to be completed by the end of September.

Contents

Preface	ii
Executive Summary	iii
Introduction	2
User Testing	3
Performance and Efficiency	4
Portability	4
Performance Testing	
Usability	6
Future Improvements	8
Update Specification	
Maintenance	
Schedule for Improvement	10
Conclusion	

Introduction

In our initial project plan, we proposed to develop and maintain a smart phone application designed to assist in the management of habits. The first version of our habit management application was successfully completed and submitted as the second deliverable (See Figure 1). Our application was designed to provide regular reminders to a user to help them manage taking up good habits, for example taking regular micro-breaks while engaged in sedentary activity. The target user for this was a student or office worker who spends a lot of time sitting down and also has other activities or interests that a customizable timer would be useful for.

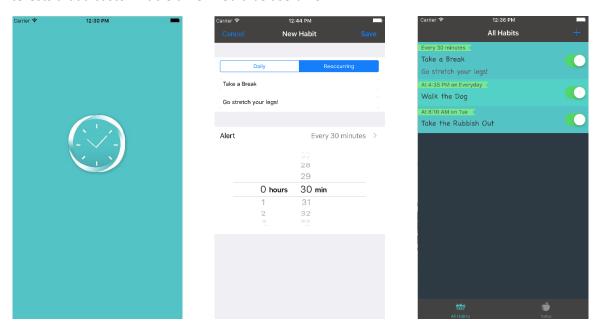


Figure 1. Screen captures from out application. The loading screen (left), creating a habit (middle), and the all habits view screen (right).

In the first version of the application, users were able to create habits that would generate a notification at a certain time and/or day of the week. The notifications consist of a description of the habit provided by the user and are handled through the iOS User Notification framework. Users could also create reoccurring timers set to go off after a user specified time interval, and then repeat. The main screen has two tabbed views; one displaying all of the habits relevant to today, and another displaying all habits.

In the design and implementation we aimed for the application to exhibit efficiency, dependability, maintainability and usability. After the first version of the application was submitting we dedicated our time to testing and determining if it displayed these attributes. This report contains analysis of user testing and feedback over the period from 1st June to 31st July 2017. The report also includes suggestions for improvement and a schedule for the implementation and testing of the new features we will be adding to the app.

¹ COSC345 Deliverable 1: Project Plan, available at https://github.com/powerofpercy/GottaHabit/releases/tag/1.0

² COSC345 Deliverable 2 : Working Software, available at https://github.com/powerofpercy/GottaHabit/releases/tag/2.1.1

User Testing

User testing of the application was carried out by a total of seven people over the period from 1st June to 31st July 2017. Of those seven people four of them were in the development team and three were external to the team. All the testers were students at the University of Otago studying full time.

Users external to the development team had the application installed on their phones and were prompt to attempt to use the app in any way they saw fit. Minimal instructions were given to the users on how the app worked in order to see if it was easy to use. They were then left to use the app and prompted for feedback at the end of the testing period. We asked them to provide open feedback in response to the following questions:

- What did you use the app to do?
- Did you find the app easy to use?
- What did you like about the app?
- If you could add or change one feature to the app what would it be?
- Did the app ever stop working or take a long time to respond?

We also collecting information about the model of iPhone the external users had and the operating system they were running.

Performance and Efficiency

Smart phones are multifunctional devices and a user may run multiple applications at once on their phone. In our design we wanted the app to be efficient so that it would not interfere, slow down or adversely affect other applications running on the phone at the same time. We also intended our application to run reliably on a number of different iPhone models. An application that doesn't run or runs slowly may well be unusable and so this is of primary concern when assessing the usefulness of the application.

Portability

Simulator Testing

In our initial specification we intended the application to work with all iPhone models capable of supporting 64-bit applications in iOS 10. This includes iPhone models from the iPhone SE to the iPhone 7 Plus. To ensure that our application could be run on all the desired models, we automated the build testing process using the Travis CI service. At every commit, our application was run on a simulator for each relevant iPhone model. To date we have found no peculiarities to any one iPhone model, meaning that if it builds on and runs on one model, it should run on all the others capable of supporting iOS 10.

Physical Testing

User testing of the software was carried out on a number of different iPhone models including the iPhone 7 Plus, iPhone 6s, iPhone 6 and iPhone SE. All of these devices we able to run and use all the available functions of the application. During testing, one of the tester's iPhones was broken. The user asked if she could use the application on her iPad. Compatibility with the iPad was not something we had considered but we may include in the next version.

Performance Testing

Responsiveness

When testing the application we were looking for any humanly noticeable delays between user action and the response of the application. External users did not report slow response times or crashes in the operation of the application nor did they notice it adversely affecting any other application running concurrently on the application.

Testing within the development team found that a few elements in the interface were slow to respond or update. When switching between the day selector and habit description when making a new habit, the selected days take a second or more to appear on the screen. There is a similar delay that occurs between selecting 'Done', to create a new habit, and the habit appearing on the main habit screen. These delays occur regardless of the model of phone we used and may represent an inefficiency in the software rather than hardware. Other habit handling functions we wish to add may be affected by the same issue and so we aim to fix this for the next version.

Efficiency

In addition to user testing we also used the Xcode iPhone simulator to monitor CPU usage and memory use for our app. Most functions of the application use a very small fraction of the CPU's processing power. Generating notifications, turning habits on and off, scrolling through a long list of habits and deleting habits all generate spikes consuming less than 3-4% of the CPU (See Figure 2). Adding habits takes a bit more spiking at 15-30%.

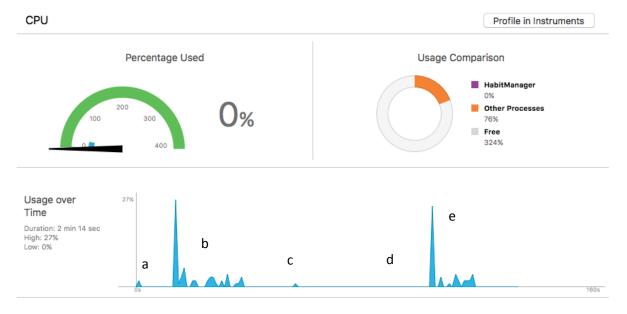


Figure 2. A typical CPU monitor screen while running the application on Xcode iPhone simulator. Different functions are visible in the usage over time; a) launch of application, b) and e) adding a habit, c) deleting a habit, d) notification sent.

In the simulator we were also able to stress test the application by adding 1024 randomly generated habits to the application and monitoring CPU and memory usage. Adding all of these habits at once used up to 60% of the CPU and consumed 300- 400 MB of memory. Once the habits were added CPU usage dropped back to normal values for scrolling, adding and deleting habits. This testing shows that excessive use of the application can still be handled by the iPhone hardware.

Usability

A common user expectation is that any smart phone application should be intuitive and easy to use. The iPhone especially has extensive guidelines for designing easy to use applications.³ In our application we tried to make the app look attractive and consistent, provide familiar elements to interact with and keep the interface clean and simple. Some of the responses from users are included here in quotation marks.

Learnability

"Works about how I expected"

All of the external users were able to use all the functions provided to them in the app without explicit instruction on use each one. The app was used for a variety of functions and each user was able to customise the habits to suit their needs. The user interface was simple with as few elements as possible. Many of the elements such as sliders and timer scroll wheels were already familiar to the users from other applications, reducing the learning curve for our application.

Flexibility

The original intention of the app was for it to be used for habits such as taking micro breaks during long periods of sitting down. Other uses for the app that our users reported include scheduling lectures and generating reminders to take medication, and reminders to check the stove when cooking.

Appearance

"It does look pretty"

We had multiple comments on high quality appearance including the logo and consistent colour scheme throughout. One user raised the possibility that different colours could be used to quickly distinguish between daily and reoccurring habits. A different colour for expired habits would also help them stand out.

Functionality

"How do I modify a habit?"

In the first version of the application we did not include the ability to change and edit habits after they had been created. The only way to change a habit was to delete the old one and make a new habit with the desired details. This was very frustrating and most users expressed dissatisfaction with it. We aim to prioritise this function for our next version of the software.

"Why is there a slider on an event that already happened?"

The application allows the users to make one off today only reminders in addition to daily and reoccurring habits. However we found that when these expired, they remained on the screen and the user could interact with them like any other habit. It would be more intuitive for these habits to be disabled after they had expired.

³ Apple Developer Human Interface Guidelines, available at https://developer.apple.com/ios/human-interface-guidelines/overview/design-principles/, last accessed 13/08/2017.

The initial user specification detailed that the app would have an option for a reoccurring timer with customisable segments of different lengths. The intention was that in the micro-break example, users could specify work and break length. We initially implemented a simpler reoccurring timer with only one length of time to repeat. In testing it was found that breaks and working periods can vary a lot for each user and it was difficult to specify the length of a period for break or work. In this context it was easier to use a single reoccurring timer and use the stop/start function to stop the timer for breaks and start it again when going back to work.

The use of notifications instead of an alarm function was also seen as a good thing by users. Notification alerts tend to be more subtle than alarms and don't force the user to attend to it immediately. This prevents the app from distracting the user if they are fully engaged in a task. Notifications are also automatically muted during calls as desired in our initial user specification.

Future Improvements

Based on the feedback from users and from testing ourselves we have developed a specification for new features that we wish to add and bugs/features that need to be improved. The new features have been grouped into key features which are high priority for the next version and extended features to be done after the key features have been implemented.

Update Specification

Key Updates

1. Edit Habits

1.1. The user should be able to change the name, time and/or days for all habits both daily and reoccurring.

Rationale: Mistakes may be made in the details of a habit and the user should be able to quickly correct the details without having to make a new habit and delete the old one. Habits may change slightly over time and the user should be able to adjust the timers to suit their own preferences.

2. Monthly Habits

2.1. The user should be able to create a habit that generates a notification on a certain day or days of the month.

Rationale: Monthly reminders will make the application more flexible and able to be used for other habits that occur less frequently than weekly. The user should be able to select multiple days if required.

3. Colour coding of habits

- 3.1. Expired habits should change colour to differentiate them from current habits.
- 3.2. Reoccurring habits should have a different colour from daily and monthly habits to help the user distinguish between the two.
- 3.3. Any colour changes and differences should not detract from the visual appearance, the multiple colours must still be cohesive to the colour scheme.

Rationale: The use of colour will highlight different types of habits and allow the user to distinguish the different types more quickly. Habits should be greyed out and become inactive when they have expired. Clashing colours should be avoid to maintain the appearance and visual cohesion of the interface.

Extended Updates

4. Sort Habits

4.1. The application should display the habits in chronological order.

Rationale: The user should be able to find consecutive habits easily without have to scroll through and manually pick out which ones happen in what order. The daily habits should be listed by day then time. Reoccurring habits may be listed together before the list of daily habits.

5. Search Function

5.1. The user should be able to search for habits based on their description, name and start time. *Rationale:* When dealing with a large number of habits the user should be able to find and interact with a specific habit quickly. A search option should be available as an alternative to scrolling through all the habits to find the desired one.

- 6. iPad Portability
 - 6.1. The application should be able to run on all versions of the iPad compatible with iOS 10 and function in an equivalent manner to the iPhone version.
 - 6.2. The user should be able to transfer habits from the iPhone version to the iPad version using Apple iCloud service.

Rationale: A port to the iPad should be relatively easy given that the iPad and iPhone use the same iOS operating system. This would allow a user to use either device available to them when using the app. If a user has both an iPhone and iPad then they should be able to sync the habits between the two different devices to prevent inconsistencies and redundant entering of data. Apple provides the iCloud service for this purpose.

Maintenance

- 7. Improve response time for day selection and display newly added habits.

 Desired Behaviour: There should be no noticeable delay between the weekday selected screen loading and the ticks marking the days being selected. Similarly there should be no noticeable delay between the loading of the habit screen and the appearance of a newly added habit.
- 8. Remove the ability to interact with expired habits Desired Behaviour: The slider on any habits that are no longer relevant i.e. the time they were set for has already passed, should no longer work.

Schedule for Improvement

The final deliverable is due on the 26th of September, approximately 6 weeks from the submission of this report. A schedule for completing the desired changes is included in Table 1 and presented as a Gantt chart in Figure 3. Work on the new features and bug fixes mentioned here has already begun and this is reflected in the schedule. Each task has been given roughly a two week block for its completion, at the end of each two week block we will meet briefly to discuss progress and reassess the schedule. Testing of new features will occur immediately after they have been implemented, in parallel with the development of other new features. A full review of progress, risks and documentation will be done four weeks before the deadline to ensure we are on track for completion.

Table 1. Schedule for complementation of the finished software product including implementation of new features and fixes.

ID	Name/Title	Estimated Days	Assignment	Dependencies
1	Maintenance & Testing	120		
1.1	Collect User Feedback	60	Whole Group	
1.2	Assess Compatibility of Software to iOS/ iPhone 8	4	Whole Group	
1.3	Analyse User Feedback, decide on new features	3	Whole Group	1.1, 1.2
1.4	Deliverable 3: Usability Report	N/A	Alexis	
1.5	Implement Edit Function	10	Percy , Tyler	
1.6	Debugging Existing Features	10	Percy , Tyler	
1.7	Implement Colour Coding of Habits	10	Percy , Tyler	
1.8	Implement Monthly Habits	10	Percy , Tyler	
1.9	Testing of Edit Function and New Colour Scheme	10	Alexis, Maggie	1.5, 1.7
1.10	Testing Monthly Habits	10	Alexis, Maggie	1.8
1.11	Documentation Inspection & Progress Review #3	N/A	Alexis	
1.12	Implement Sort and Search Function	10	Percy , Tyler	
1.13	Implement Support for iPad and iCloud	10	Percy , Tyler	
1.14	Testing Sort and Search Functions	10	Alexis, Maggie	
1.15	Final Testing and Debugging New Features	10	Whole Group	
1.16	Deliverable 4: Finished Software	N/A	Whole Group	1.15

teamganttCreated with Free Edition

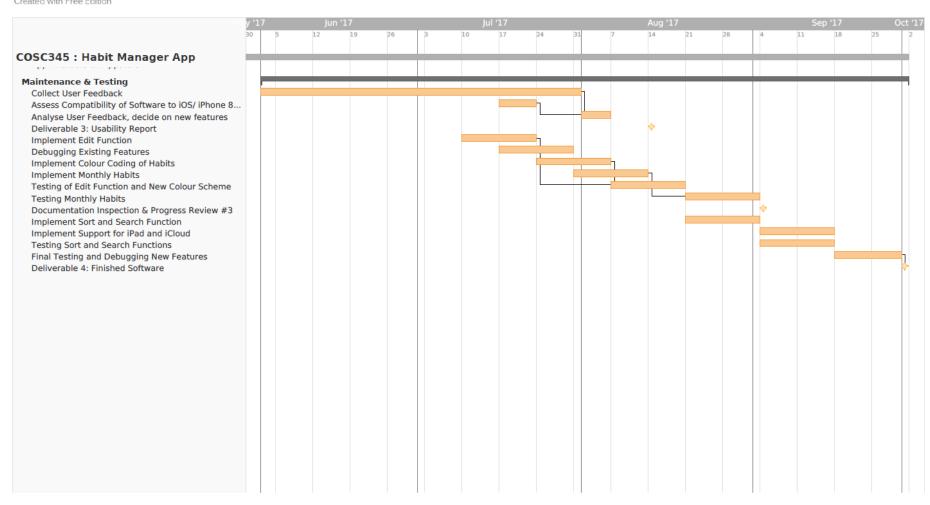


Figure 3. Gantt chart for the completion of the software product.

Conclusion

In general, the users testing our habit manager application found it easy to use, attractive to look at and useful for a variety of different habits. It was clear, however, that having an edit function was a much needed feature for the application. Other features were also suggested such as colour coding of habits, monthly habits and compatibility with the Apple iPad. We plan to implement a number of these beginning with essential features such as the edit function. Time permitting, we also plan to add more extended features such as search and sort functions, and if possible port the application to the iPad. Continuous testing and evaluation will continue in parallel with implementation to ensure the quality and usability of the software does not decline. The result of these changes will be a final working and debugged version of the software ready for the fourth deliverable.