

Diabeatit SP19 User Testing Report

Introduction	1
The Script	1
User Testing In-Action	2
Results & Interpretation	3
Discussion for Future Teams	4

Introduction

What is This and How Can it Help?

This User Testing Report from Spring 2019 details the first brief guerilla user testing done for the new Diabeatit UI (started over from scratch in Winter 2019). The script the SP19 UX team followed will be provided as well as an example script for future testing. Recorded observations, thoughts, and interpretations will help explain some of the decisions made for the (as of now) current UI prototype on Figma. The document will leave off with a discussion on how to account for potential problems the UI may face in the future and what future teams can do to address them.

The Script

Context

Due to organizational issues and a fast-approaching deadline, user testing was done on feedback-aggregated lo-fi paper wireframes. Three random participants from the Diabeatit team were chosen and asked to perform two specific-tasks within the UI.

Rationale

We used the specific-task-based testing method based on a [DSGN 100 worksheet](#) for quick and dirty user testing. The five-second test was preferred, but due to time and lack of proper preparation was held off for future testing. The DSGN 100 worksheet's tools were chosen for their effectiveness in quickly obtaining insights into UI breakdowns, navigation issues, etc.

Template

Our session involved two roles, one facilitator and one note-taker. The *facilitator* spoke with and guided the user through the testing process by asking them to perform certain tasks, answering any questions, and eliciting feedback and opportunities for discussion by encouraging them to think out-loud. The *note-taker* recorded moments where the user discovered errors, implicit expressions or gestures made that indicated a clash of mental-models, and the feedback and discussion shared with the facilitator. As for the structure:

Greetings and Notes:

- Politely greet participants and thank them for their time.
- Briefly go over the tasks they'll be doing and how doing so will help the project.
- Encourage them to think aloud during the test.

Task(s): (Refer to the [DSGN 100 worksheet](#) for more details)

- 5 Second Tests
- Specific Task-Based Tests
- During these tests, make sure the facilitator is frequently drawing out information from the user and the note-taker is frequently recording said information.
- **Notes:**
 - Your needs may vary. For SP19, we only used the specific task-based tests due to time constraints.
 - While the note-taker can talk with the user, they should focus on recording information in the background to ensure no details are unnoticed.

Discussion + Follow-up Questions:

- This is both the team and users' opportunity to clarify issues and interesting actions that came up during the tests.

User Testing In-Action

Observations and Records

Before the sessions, we developed two main task goals that users should be able to accomplish. The goals are as followed:

- Get to and Play Kitchen Showdown
- Accessing Instructions and Educational Content

During each of the three sessions, the facilitator and the note-taker first greeted the participants and briefly went over the project and the task that they will be doing for the user testing. We, then, encouraged them to think aloud throughout the testing, before starting the test. Each user testing lasting about ten minutes.

The *facilitator* briefly introduced the prototypes to the participants and gave instructions and the user tasks. The facilitator showed the prototypes by order and allowed the participants to

interact with the prototype freely. If participants have any questions during the user testing process, facilitators can also answer their questions. However, participants should try to go through all the user tasks while the note-taker can take note about his/her observations.

The *note-taker* observed the users' verbal and non-verbal interactions with the prototype during the two tests. After the facilitator instructs each of the tasks, the note-taker began by observing the user's initial hand and eye movement as they interact with the prototype. For instance, the user may have fixated their eyes on a single spot of the page, but their hands may be hovering over the entire screen page. At the same time, the note-taker also took note of what the user is verbally expressing and if the verbal and non-verbal cues correlate with one another. The note-taker also observed how long it takes for the users to get familiar with each screen page.

Results & Interpretation

What Happened and Why

From the three user testing, we were able to gather the following results and interpretations:

- Character Customization:
 - **Result:** User looked at the bottom of the customization screen and verbalized that she thought all of the customization would happen on one screen. User also clicked the "Okay?" button after each customization screen.
 - **Interpretation:** The customization bar on the bottom of the screen lack the affordance that each of the customizations (skin color, hair color, etc) will happen individually. The placement of the bar on the bottom also limits the user's ability to quickly visualize and understand what specific customization is happening.
- Introduction:
 - **Result:** Users are confused about the arrows' functionality in the intro story screens
 - "What happens if you press the right arrow on the cookie page [with additional selections]?"
 - In the first introduction story screen, "what does the left arrow do?"
 - User verbalized that he thought the right arrow would lead to the home game screen.
 - **Interpretation:** The left and right arrow signs on each side of the dialogue boxes of the introduction screens are supposed to lead to the previous and next screens, respectively. Due to the size and the placement of the arrows *outside* of the dialogue boxes, its functionality is not conveyed clearly to the users. Furthermore, on interactive screens, the users are given multiple choices that would lead to the same results, causing additional confusions.
- Homescreen:
 - **Result:** User was surprised and took time to look through the house before selecting Kitchen Showdown on the home menu. User verbalized that she is not sure if the screen with the house is the home screen.

- **Interpretation:** The structure of the home screen lacks the clear affordance to the users of its purpose.
- Kitchen Showdown:
 - **Result:** User was confused if the back button in level 2 would return to level 1 or the homescreen. User was not sure what would happen when she skips the difficulty level page.
 - **Interpretation:** The users are given multiple choices that lacks the affordance of where each choice would lead to. There should be a clear distinction and affordability for each of the buttons on the screen.

Discussion for Future Teams

While designing the current Diabeatit UI, the team ran into some potential issues we'd like to discuss with future teams. We hope that they can build off the work and insights we discovered during the quarter to create consistent progress for Diabeatit.

Problems to Consider

- **Accounting for Multiple Game Loops**
 - Most games consist of one core loop that's utilized across the entire program. In Diabeatit's case (at the time of writing) it's intended to be a collection of diverse game loops that revolves around the theme of diabetes education. While this can help diversify gameplay on the surface, meaningful depth may be lost.
 - Until now, Kitchen Showdown was the main area of focus for development, with UX and Storyline switching between Kitchen Showdown and the overall narrative and UI. We highly recommend future teams to design mini-game concepts to help guide UI and narrative development.
 - Helpful sources of information may include:
 - Party Games:
 - [Super Mario Party](#)
 - [The Jackbox Party Pack](#)
 - [Carnival Games](#)
 - [Wii Play](#)
 - Collections of micro-games (3-5 sec. games):
 - [WarioWare](#)
 - In all examples, you will notice that each mini-game or micro-game is intended for relaying brief but intense moment of high-energy. The problem one could face with designing for an educational game is attempting to give players important information about diabetes that may require more time than a brief mini-game. The current workaround is to incorporate multiple levels of difficulty and have players repeat the loop until they understand the material. Future teams may find different methods that fit the game's educational needs.

- You may also notice that all of these examples don't have much in terms of a story. SP19 wanted to try incorporating more story elements, but we realized that doing so could just bloat the game. Future teams can choose to rewrite it how they will.
- **TL;DR**
 - **How can we teach players about diabetes by designing for multiple game loops?**
 - **How can incorporating a story be helpful towards a stronger user experience? How can it also harm the user exp.?**
- **UI Scalability/Flexibility/Durability**
 - Following the issue with accounting for the possibility of multiple game loops, the ability for Diabeatit's UI to account for said loops comes into question. Since we didn't know how many mini-games would be included later on, the SP19 team designed a temporary UI that allows for a basic scrollable 3-column grid. The main issue is we don't know if this is viable for holding up in the future. The 3-row grid can be used, but whether it's the best way for users to access mini-games is for testing and further discussion to show.
 - **TL;DR**
 - **How can players best access the mini-games in the UI?**
 - **How can "best" be measured? By efficiency (time, navigation, etc.)?**
 - **How can we design a container that is scalable (accounts for varying amounts of mini-games), flexible (does not require hard-coding or many edge-case handling), and durable (can hold up under sudden or impactful changes)?**

Ideas to Consider

- **Splash Screen**
 - If the game ends up taking some time to load, a splash screen may help leave an impression on players in the meantime. [Take a look at the benefits.](#)
- **UI Heuristics**
 - [UI Heuristics as applied to video games](#)
 - When looking at the current UI, it may help to keep Nielsen's 10 UI Heuristics in mind.
- **Diegetic Game Elements for Better Immersion**
 - Diegetic elements are a great design tool in games for presenting narrative, immersion, and more intuitive UI's (potentially). Here's [a great analysis](#) of diegetic elements in a number of games.
- **Physical Limitations**
 - Consider UI layouts that can accommodate small hands (or even only one hand).

Helpful Resources

- ***Designing Games for Children***
 - [Effective Use of Color and Graphics in Applications for Children, Part I: Toddlers and Preschoolers](#)
 - [Designing For Kids Is Not Child's Play](#)
- ***General Game UI***
 - [Video game UI: The design process explained](#)
- ***Usability Testing***
 - [Quick and dirty usability testing](#)
 - [Usability Test, Even When You Know the Answer](#)
 - [How Many Test Users in a Usability Study?](#)
 - ['But You Tested with Only 5 Users!': Responding to Skepticism About Findings From Small Studies](#)