

# FunPod: Medium Fidelity Prototype Write-Up

Link: <http://invis.io/XT1M31R4Z> or <http://www.henryhtran.com/#/funpod/>

## Problem and Solution Overview

For many children with diabetes, managing their condition is very scary and daunting. Diabetes care can be a flurry of self-administering insulin shots, pricking and testing blood, visiting the doctor, etc. Our hope is that by minimizing these fears and simplifying these processes, children can be more relaxed and learn how to manage their health better. Currently, many children manage their diabetes through a Personal Diabetes Manager (PDM), which is a device that tests blood glucose, communicates to an insulin reservoir, and visualizes blood glucose data. Currently PDMs are not user-friendly, and definitely not kid-friendly as the interface is not intuitive. We are improving the Personal Diabetes Manager by designing a kid-friendly interface that integrates reminders, metrics, educational games, and positive reinforcement.

## Tasks

The 3 tasks we want our application to support are:

1. Checking blood glucose (simple)

This is a simple task that takes less than a minute for people who have had diabetes for several years. It is also done very frequently (7-8 times a day), usually before meals, before bed, and whenever one feels ill. Our app assumes that the user has a Continuous Glucose Monitor System, which would constantly measure blood glucose. However, users do need to double-check their blood glucose level at least 1-2 times a day. Our app has a quick way of allowing for this, assuming our device has the hardware of existing PDMs.

2. Counting carbs/learning to count carbs (moderate)

Counting carbs requires quite a bit of practice. In fact, it was one thing we found caused anxiety in diabetics because it can be inaccurate and currently involves looking up/memorizing of the number of carbohydrates in different foods. Therefore, our app includes a carb counter that supports both manual entry of carbs through a database as well as a feature that allows the user to take a picture and instantly get a carb estimate.

3. Administering insulin (complex)

Administering insulin is complicated because you have to consider a variety of questions such as: "What have you eaten?", "What was your blood glucose previously?", "What activities will you be doing later?", and finally "What amount of insulin do you need?". Our application asks the user questions that they can answer to get an estimate of how much insulin they need. In the event the user has eaten something, this task links in with the second task because the user can choose to use the carb count from the carb counter to get an estimate for a bolus.

## Revised Interface Design

Based on our low-fi testing, we made changes to what each icon means. One of participants in the user tests got confused in carb counter portion of the app. He thought having a check-mark as an icon to go from the carb counter to the home screen rather than the bolus screen was misleading. The participant saw bolusing as the next logical task after counting carbs and would expect a check-mark, a signal to say “done”, to lead him there. Another participant was confused about taking a picture to count the number of carbs in a meal. To remedy this problem, we have added a popup that tells the user that they are supposed to take a picture.

In addition, we changed how users can toggle between graphs of blood glucose data. Originally, we thought it was sufficient to tell users that they can use the pinch gesture to change scope. However, one of our participants suggested that having a tab bar controller of sorts was more intuitive. Thus, we implemented this in the statistics portion of our app, though we were initially afraid that having too much text on the screen.

We also tried to make the application more kid-friendly in general. For example, we replaced the term “bolus” with “insulin.” The settings page also now requires a parent to enter a passcode to change numbers like basal rate. This is meant to prevent younger children from giving themselves incorrect doses of insulin.

Most people were confused by the button labeled “fun.” To remedy this, we replaced the fun button with the more descriptively labeled “Buddy Central” button that takes the user to a menu where they can dress their buddy, play a game, and buy new accessories with the coins they have earned through taking care of themselves.

### ***Summary of New Task Support:***

We have made it so users can manually check blood sugar by accessing the trends first (shown in Figure 1) and then clicking on a button to recalibrate their CGM. Furthermore, he or she will also get a more accurate reading since glucose readings from the subcutaneous fat tissue lags behind blood glucose by about 30 minutes.

Our carb counter (shown in Figure 2) allows the user to take a picture of the food and get the carb count for each item. The user can delete items if he or she does not want to bolus for it or the app incorrectly detects the item. In addition, the user can manually enter in what they ate using the database and can add his or her own items to the database with carb count information.

The bolusing aspect (shown in Figure 3) of our app asks the user if he or she is about to exercise and if he or she is about to eat. It will give a suggestion for insulin based on these factors and will allow the user to make manual adjustments.

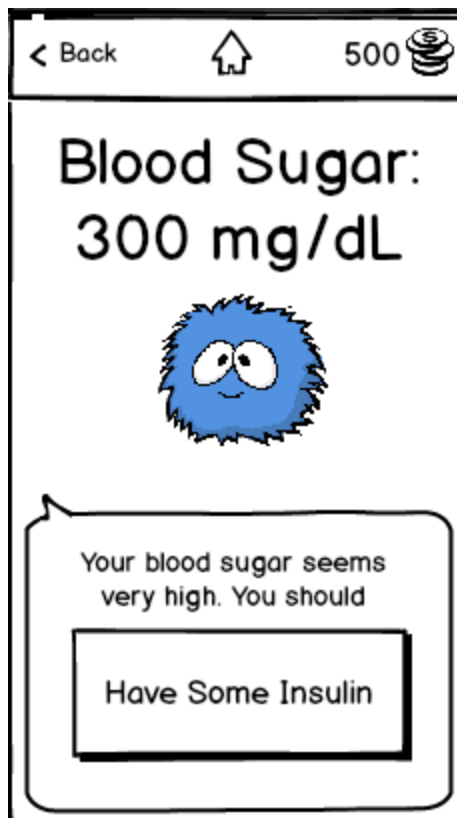
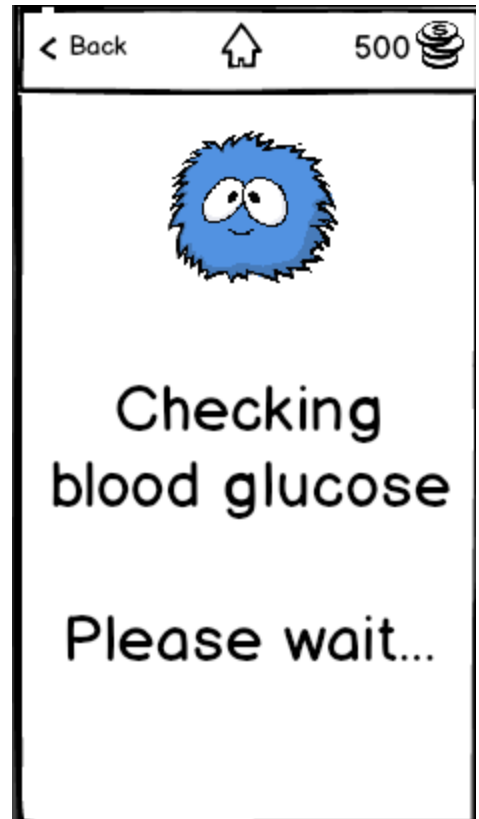
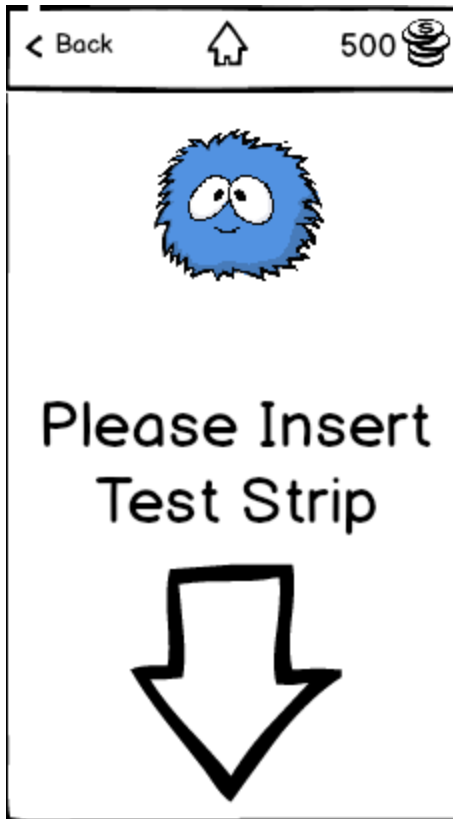
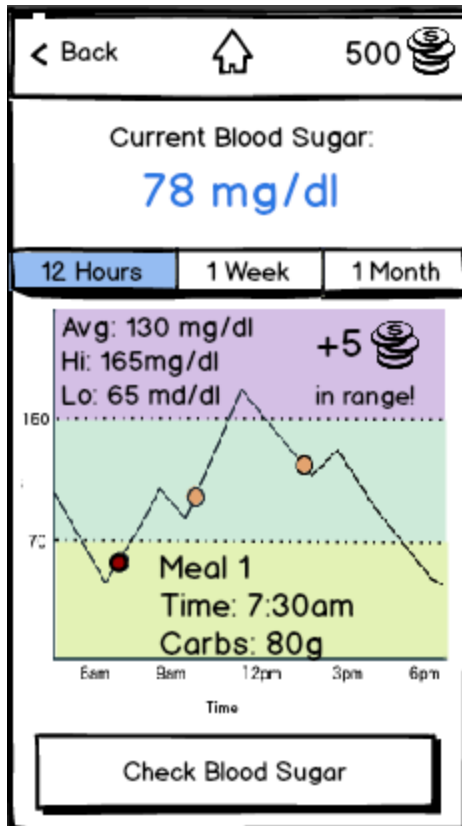


Figure 1 - Testing Blood Sugar

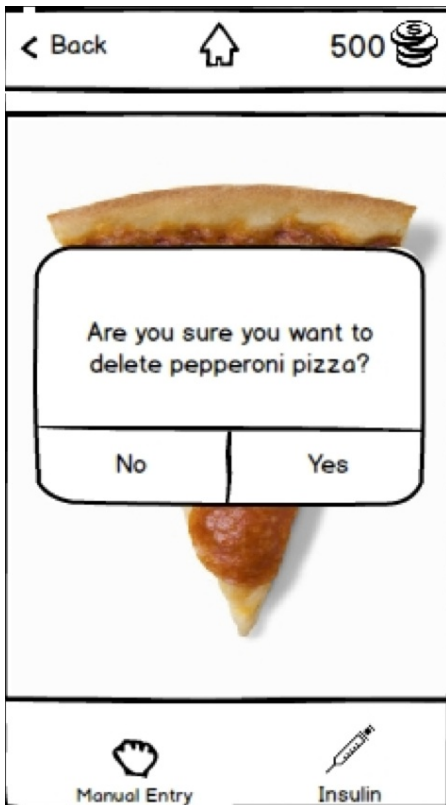
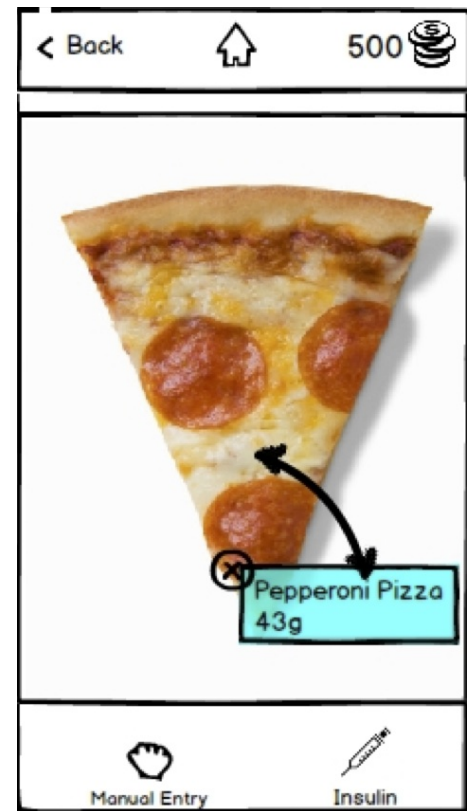
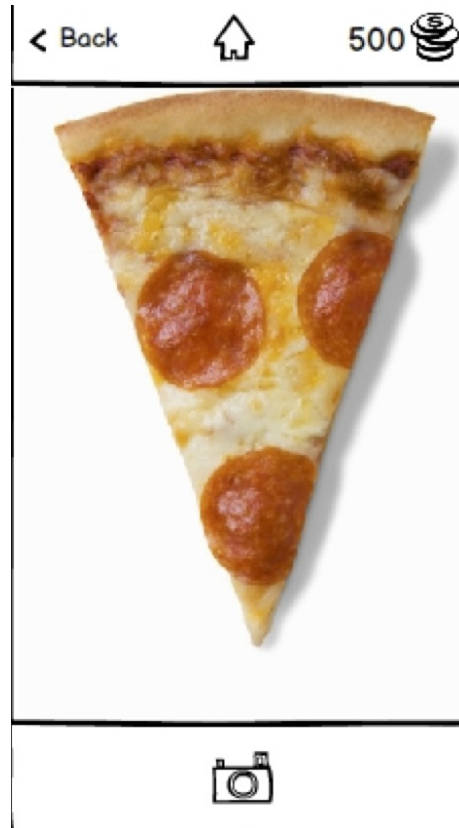
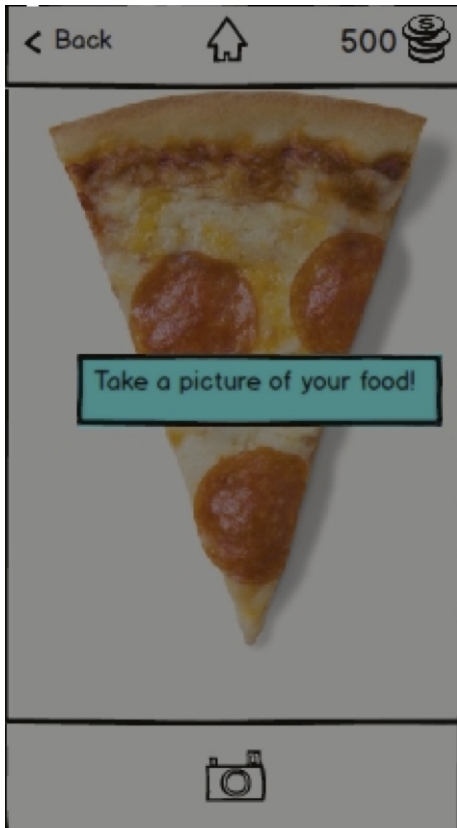


Figure 2 - Set of Carb Counting Screens



< Back 500

**Insulin**

Are you about to:

Exercise?

Eat?

I'll enter it Continue

< Back 500

**Insulin**

Are you about to:

Exercise?

When? 1 hr

How hard?  Light  Medium  Hard

I'll enter it Continue

< Back 500

**Insulin**

You should have:

**3 units**

of insulin because:

- You're about to exercise
- You recently ate 39 grams of carbs

I'll enter it Continue

< Back 500

**Insulin**

**3** units

Continue

< Back 500

**Insulin**

**3**

units of insulin

**Is this Right?**

Change  Give Insulin!

< Back 500

**Insulin**

**Giving**

**3** units

**Of Insulin**

**Stop!**

Figure 3 - Bolusing

< Back



500 



# Insulin

Stopped. Gave

**0.5** units

Of Insulin



**Continue**



## Prototype Overview

We used Balsamiq to create our app screens along with some simple graphics made in GIMP. We chose Balsamiq because it allowed us to quickly create simple screens with a kid-friendly feel. The downside of this however is that the style of Balsamiq is very distinctive and the tool does not allow for much flexibility when it comes to aesthetic choices (fonts, outlines, etc.). To link together all of our prototype screens, we used InVision. InVision allowed us to get a feel for what using the application would be like. The tool is also fairly good at accommodating changes made to the application after you have linked up several of the screens. One thing that InVision does not seem to support is animation, which made it difficult to represent some of the more dynamic portions of the app. It is also difficult/impossible to update numbers (like how many coins the user has) because you would need to create a new screen every time the number changes.

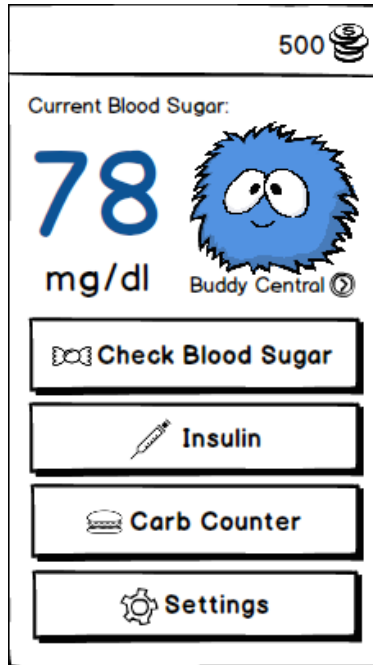
As of now, the user can only accomplish a small set of specific tasks with this prototype. We do not have data from a continuous glucose monitor, carbohydrate count database, or a camera to use to count carbs, so we have just taken some example numbers to give an idea of what the app is supposed to do. Options to change the color and accessories of the “buddy” are also not supported in this prototype because it would have involved creating many different versions of any screen where the buddy appears. In general, we set aside some of the trickier implementation details in order to produce a prototype that gives a feel for what each part of the app does.

## **Appendix**

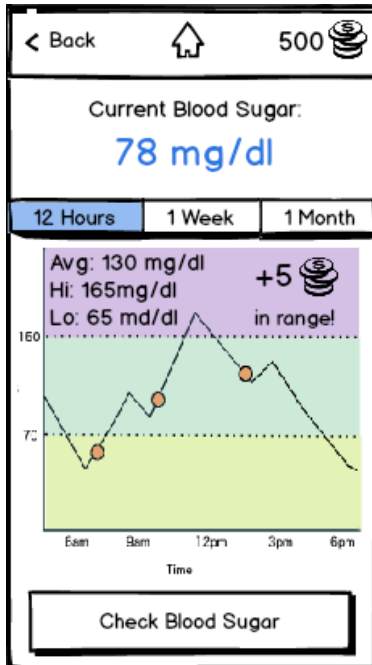
### ***README***

This application is currently running through InVision, a prototyping tool. Screens have designated hotspots that if clicked/tapped, will take the user to another screen. To navigate through the app, simply click/tap on the buttons. It should be noted that this prototype currently does not receive actual data from a Continuous Glucose Monitor, camera, carbohydrate count database, or traditional blood test strips. Many of the options presented on the screens have also not been implemented yet.

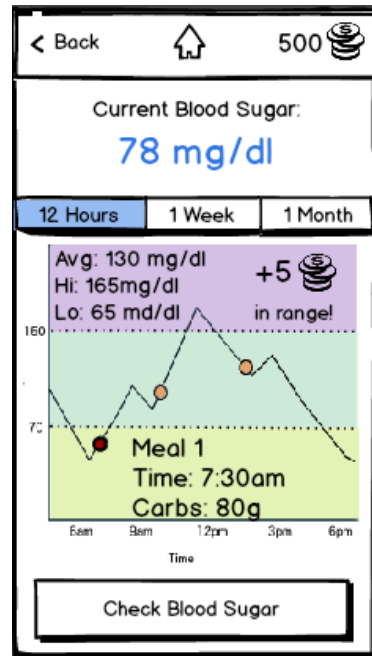
# Prototype Screens



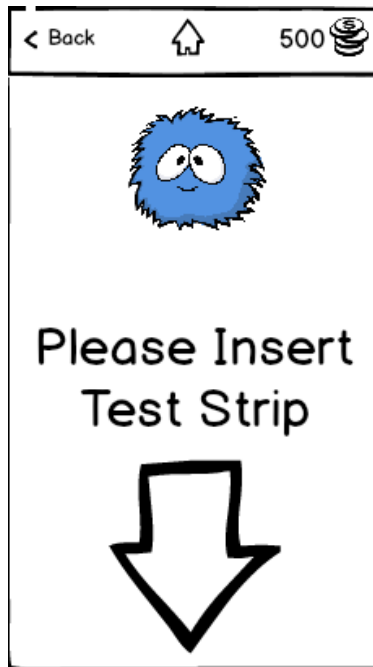
Home Screen



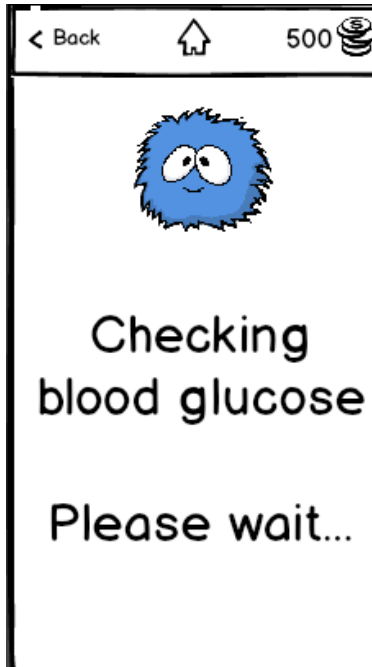
Blood Sugar trends



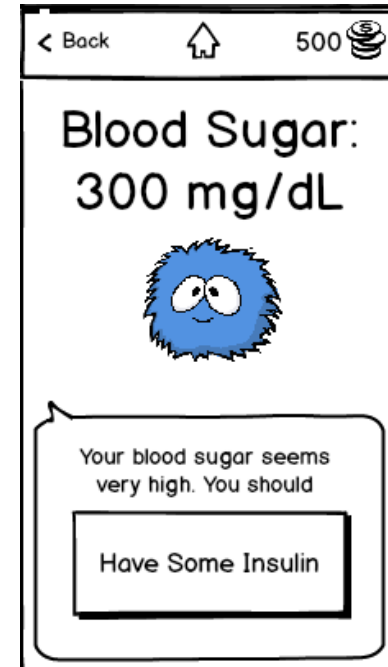
Trends with meal info



Testing BG with strips



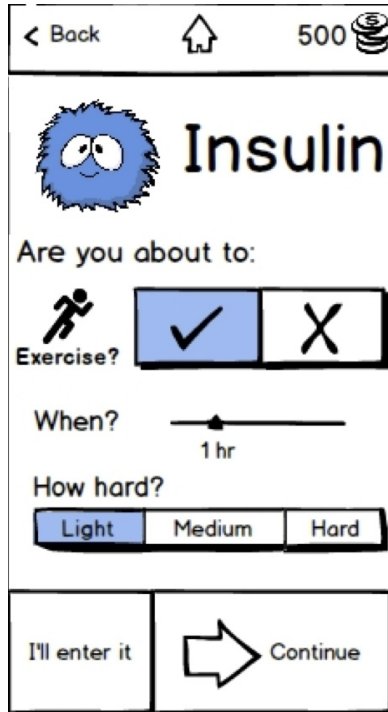
Wait screen



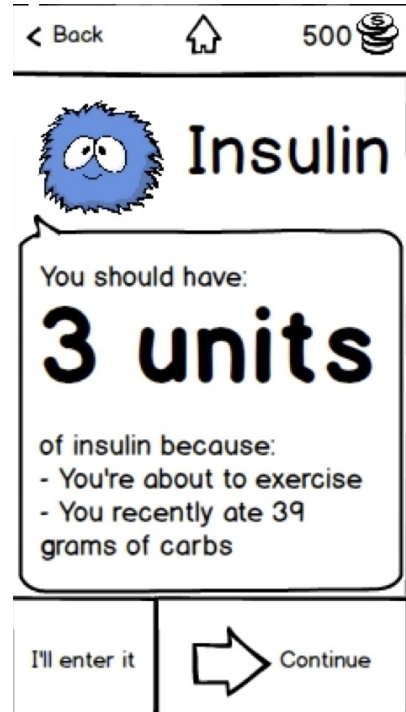
Reading with suggestion



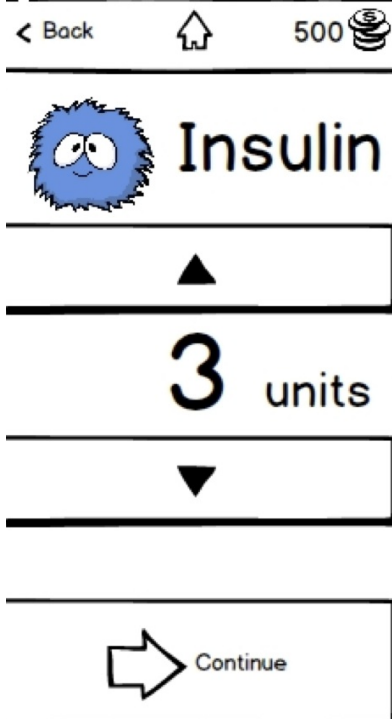
Insulin start screen



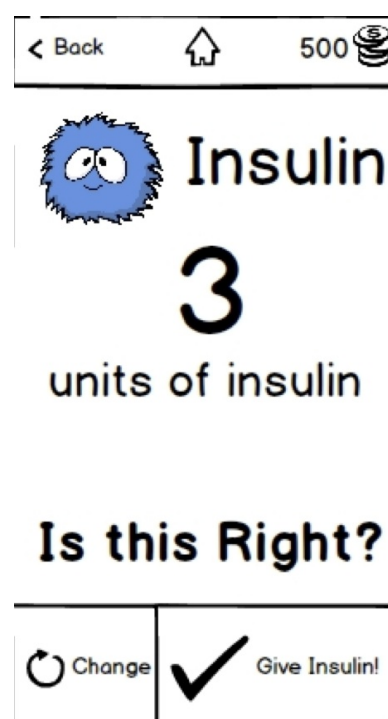
Exercise input



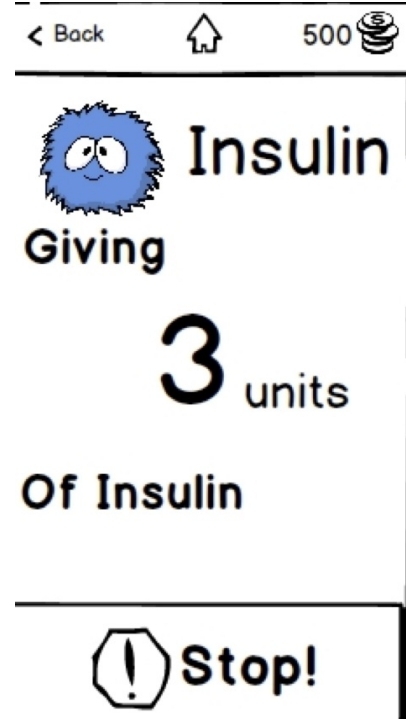
Recommended units



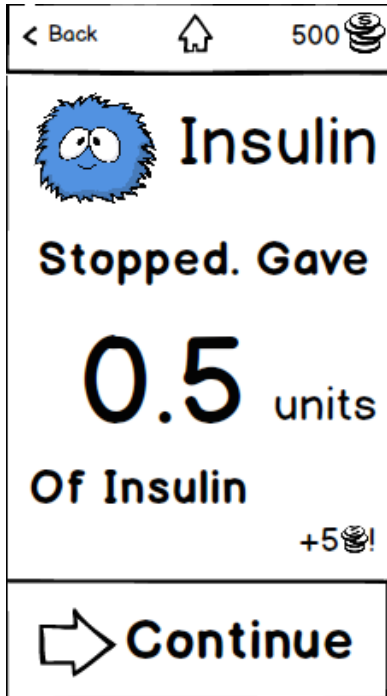
Manual adjustment of dose



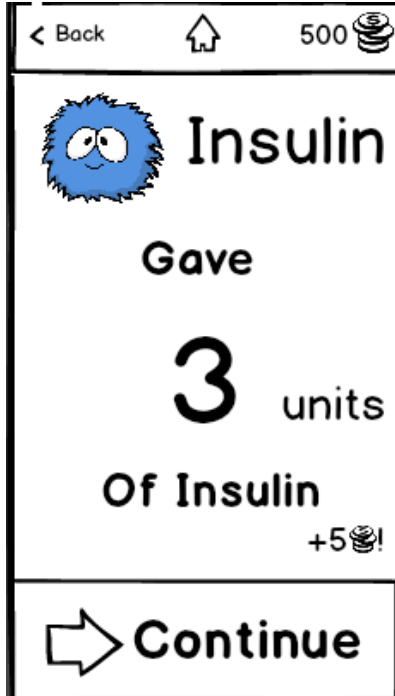
Confirmation



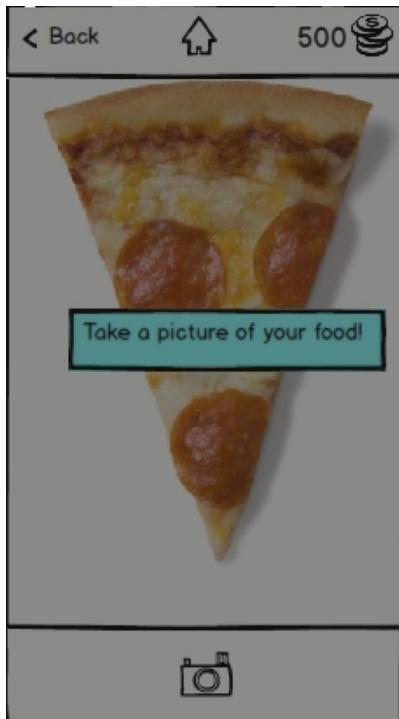
Giving insulin with option to stop



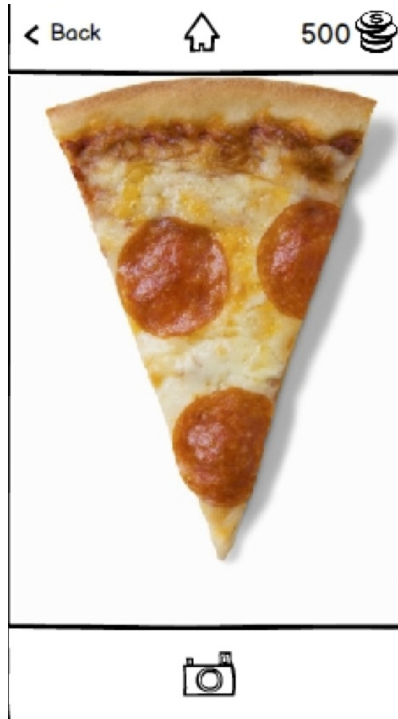
Stop screen



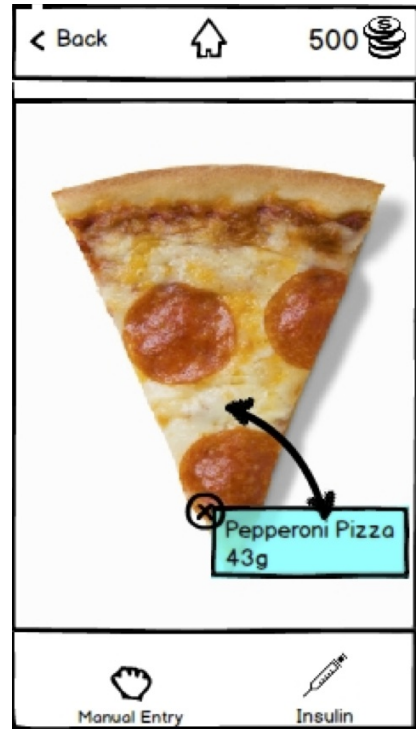
Completed bolus screen



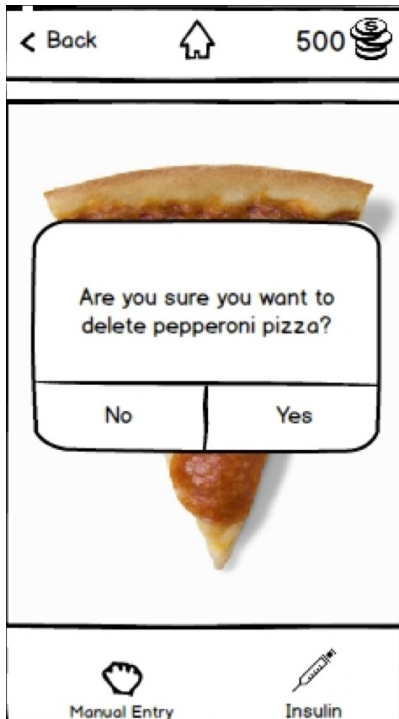
Carb counting instructions



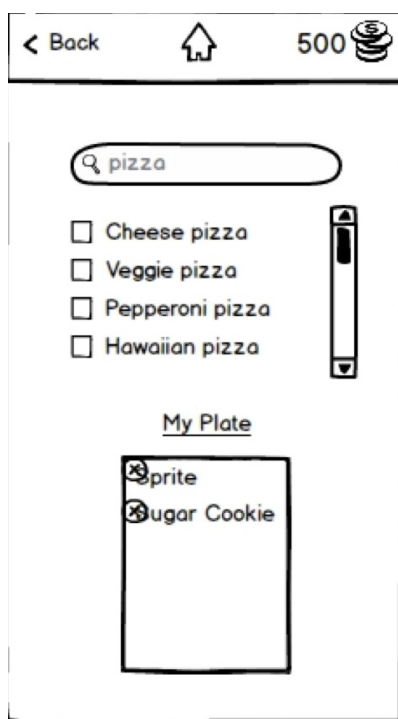
Taking picture



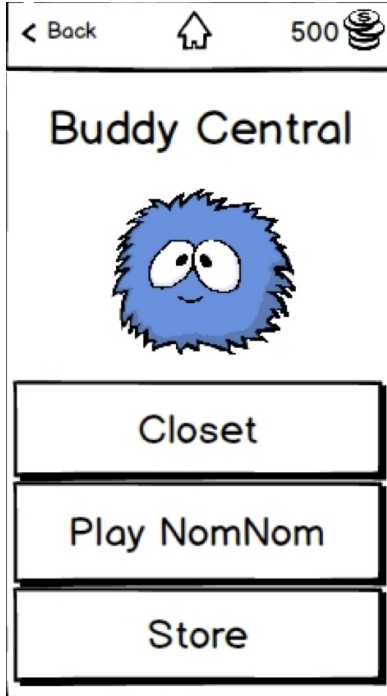
Picture with carb annotation



Delete carb count from picture



Manual carb lookup/entry



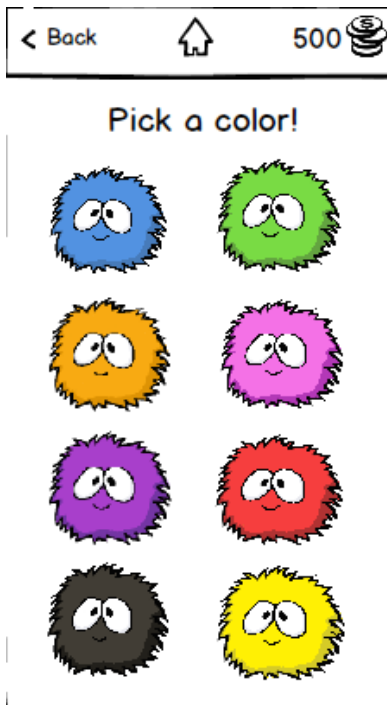
Home for games/accessories



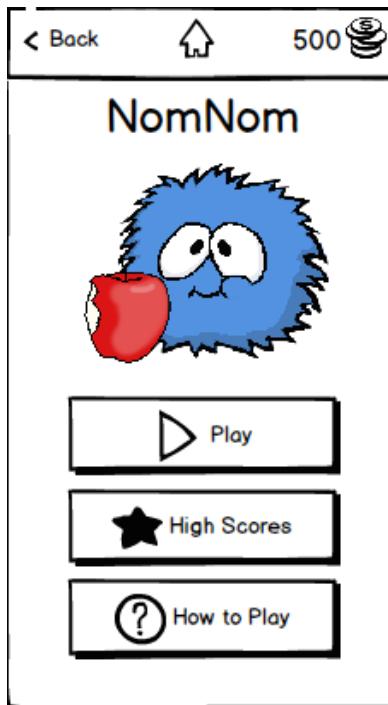
Purchased items/colors



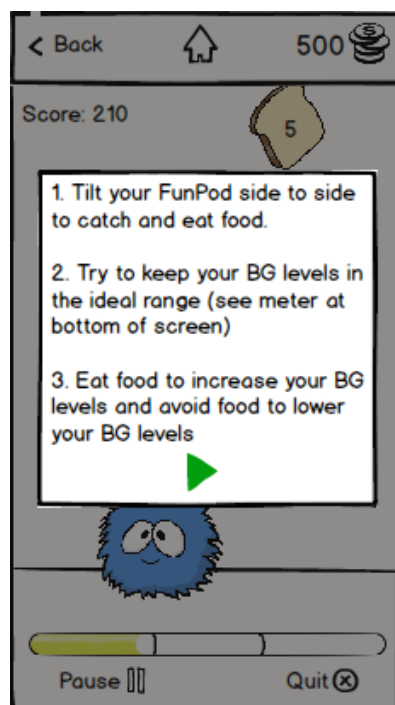
Choosing accessory



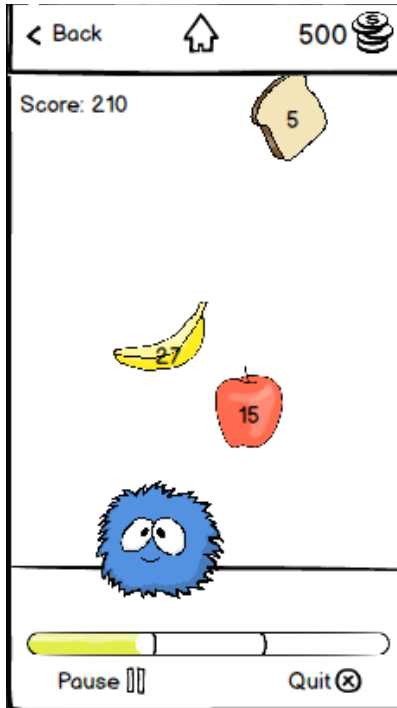
Choose color



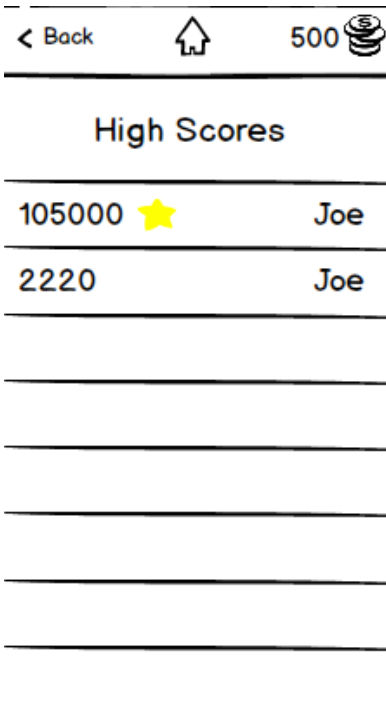
Arcade game with character



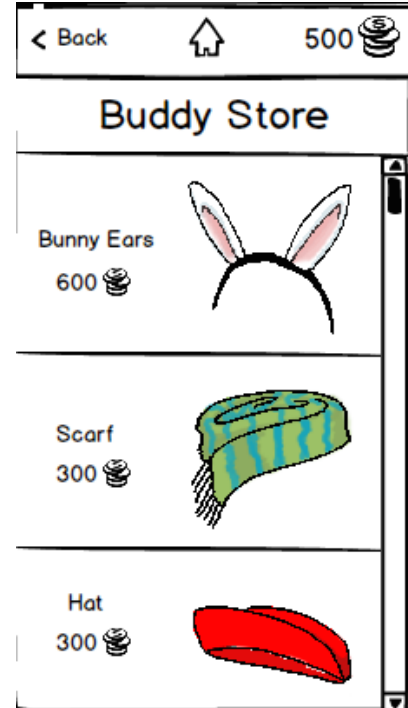
Game instructions



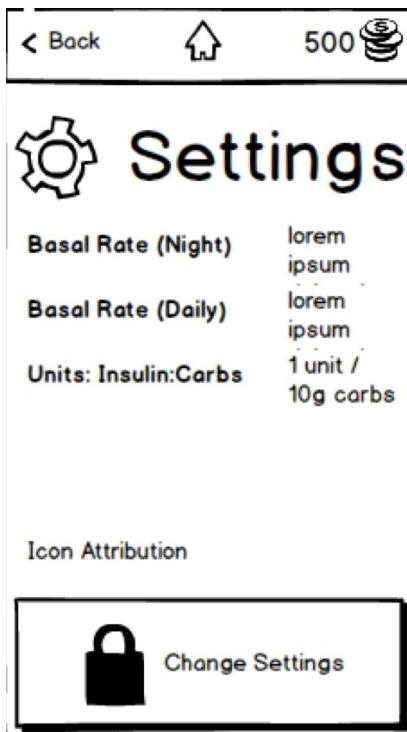
Playing game



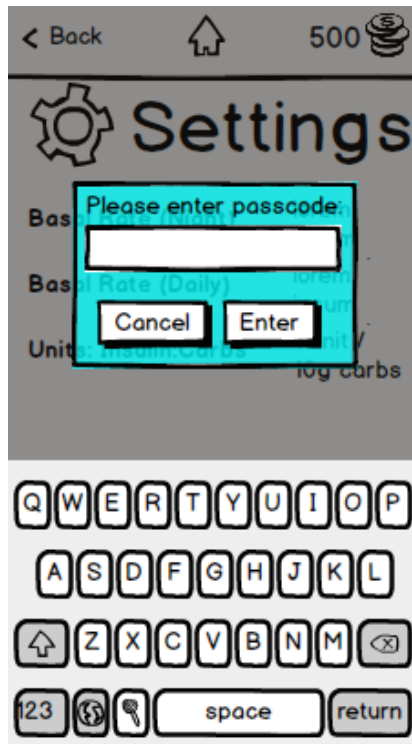
High scores



Buddy store (spend coins)



Settings, passcode required



Passcode entry