

BeanPay: Bridging the Gap between Spending and Saving.

Problem and Solution Overview

With BeanPay we set out to solve a twofold problem – Firstly, improve the current payment system, and secondly incorporate purchase data to give users the positive and negative reinforcement they need to make sound budget decisions. While this is a twofold problem we believe it is two sides of the same coin and BeanPay is the missing link.

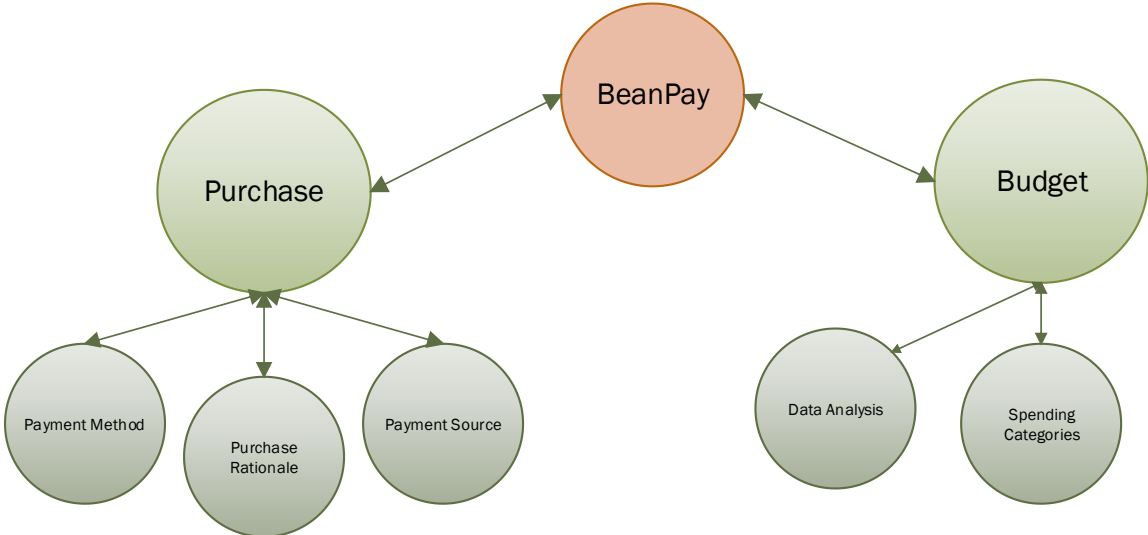


Figure 0: BeanPay

Based on our contextual inquiry we found that often there was a disconnect between the users spending habits and their budget. Figure 1 shows the relationship between purchases, budget and BeanPay. On the purchase side there are three main factors – 1) Payment method e.g. cash, credit, electronic, 2) Payment source e.g. savings, reimbursement, and 3) Purchase Rationale e.g. need, want. Whenever a user sets out to make a purchase they firstly rationalize the purchase based some existing constraints. Once they have decided to make the purchase, payment method and source will quickly fall into line.

On the Budget side there are two primary components - 1) Spending Categories, and 2) Data Analysis. After a user makes a purchase they will eventually consult their balance sheet. While this activity varies between users, those who attempt to maintain a budget will categorize their spending and analyze their spending habits based on a pre-defined budget.

The recurring theme that we observed during our investigation was that budgets are often kept mentally, and often abstractly (no hard numbers) – and then verified

at a later date. The impact of saving goals was negligible – at *best* it consisted of “shoot, I spent too much last week/month, I shouldn’t make this purchase today”.

BeanPay is an interface that incorporates various payment methodologies with simple budget analysis to seamlessly combine the purchase experience with the goal of improving a user’s spending and saving habits with real time status and feedback.

Contextual Inquiry Participants

For our contextual inquiries we primarily focused on online shoppers. This gave us the ability to sit down with a typical user and discuss the purchasing process – including why they had decided to make the purchase, as well as how they were tracking it against their budget. We took the traditional contextual inquiry role of an apprentice learning about the user and the methodologies they used to complete their purchases.

Since questions regarding financial data can be somewhat intimidating, our contextual inquiries were limited to close friends. As such the primary demographic was 25-30 year old, working professionals. While the BeanPay interface is targeted at anyone desiring to set and keep a budget, our participants were somewhat homogeneous. This gave us the benefit of identifying patterns within a given population, but also narrowed our inquiry results more than we would have liked.

Participant #1

Participant #1 is a male in his early 30’s. He is a software engineer working at Amazon. The inquiry took place in my home at the dining room table. He brought over his laptop, connected to wifi and proceeded to make an online purchase for whiteboard markers. Throughout the inquiry I maintained the apprentice role attempting to learn the physical steps involved in making the purchase as well as the psychological thought process behind the purchase.

Participant #2

Participant #2 is a male in his late 20’s. He works at Microsoft as a Software Developer. The inquiry took place in his office at work. We set aside 30 minutes so I could observe and ask him questions regarding an online purchase he was making for a reference book. Again, I maintained the role of apprentice asking questions regarding the physical as well as the mental process he went through in the steps he took to make the purchase.

Participant #3

Participant #3 is a female in her late 20’s. She works as a consultant. I walked with her through a few clothing stores, one of which she made a purchase of a cape coat. I acted as an apprentice and learned the decision making process she utilized before, during and after the purchase. Afterwards, we had a conversation about her methods of tracking her spending, what her savings goals were and how she tried to meet them.

Participant #4

Participant #4 is a male in his late 20's working as a designer for a top tier website company. In apprentice mode, I watched him purchase a movie ticket and then popcorn at the movie theater while asking questions that made the refreshments salesman look at me funny. After the purchase, I talked to participant #4 about his spending habits, budgeting techniques and his savings goals and learned how he managed his finances.

Contextual Inquiry Results

There are multiple types of payments that people make on a daily basis – necessities such as bills, groceries, leisure, hobbies, etc. When choosing our contextual inquires, we decided to focus primarily on discretionary purchases and as such provided us with a good forum for budget discussions.

As the apprentice we focused our questions around the thought process that led to the purchase, as well funding sources and methodology. Since we are trying to solve the problem of managing a budget while streamlining the payment process our questions revolved around these areas. While we did in fact take notes on the various steps the user went through while making a purchase, we were very interested in what led them to decide to make the purchase. Then when they make the purchase we asked questions around fund source and payment method. After the purchase, we discussed how they felt about the purchase, how they tracked spending, what their savings goals were, and how they met those goals.

There were several loosely connected themes that emerged throughout the inquiry process. 1) Disconnect between budget and payment process, 2) Fund source variations, 3) Interest in saving money but not in doing the work required to track it. While all of our inquiry candidates had methodologies for tracking their spending, none of them followed a strict budget. As such, the payment was one activity, and looking at their account balance at a later date was a separate and independent activity. Online purchases have a requirement that a credit card be used as opposed to using cash. With these purchases the candidates explicitly called out the card they were using and the reasoning behind it. One candidate used a particular card because it paid 5% cash back. One of the other candidates used an Amazon specific card and paid for his entire purchase using credit card points.

One of our candidates was making a purchase that would eventually be reimbursed by his company – thus his purchase would not impact his monthly budget. Since this particular candidate made over 20 thousand dollars in purchases reimbursed by his company it is crucial that a payment and budgeting solution would incorporate this scenario. His comment regarding reviewing his monthly statement was that seeing

these expenditures made the numbers “just silly” in its relevance to his monthly budget.

In addition to payments that will be reimbursed are payments made with cash. While they are totally different in nature they pose a difficult tracking problem – user input. Electronic payments can transmit payment data to a device, but certain types of payments will need user triage – this needed to be a primary concern in designing our interface: make required user interaction simple.

Bridging the disconnect between purchases, understanding spending habits, and saving money is a need we found in our contextual inquiries. Multiple candidates said that often they were surprised by the amount of money they have spent when looking at their bank statement. They said that this invoked a guilt response that would have a positive impact on their spending habits for an indeterminate amount of time. In some candidates, discovering that they had not spent as much as they thought they had was an encouragement for them to spend more in the future.

It was interesting to see that once a habit of spending or saving was established it was easy to maintain. Thus, we wanted to ensure that positive feedback was readily available through the BeanPay interface we were going to be building.

After concluding our contextual inquiry and noting the emerging themes we found our ideas for our interface evolving to adapt to the themes we saw. We originally wanted to add elements of gamification to our interface, but these ideas were not as relevant to a payments and budgeting system as we had originally thought. What was relevant was flexibility, simplicity, and enabling the user save money without imposing too much guilt.

Task Analysis Questions

Who is going to use the system?

Anyone who makes purchases and is looking to stay on budget. People who want to improve their spending habits. Someone who wants convenience in paying for things, or convenience in tracking their spending. People who just want a neat thing to pay with.

What tasks do they now perform?

Deciding to purchase or not. Decide what to purchase.

Purchasing items. Sometimes online, sometimes in person, sometimes by phone. Credit Card, Check, Cash. Pull out their CC and hand it to someone to swipe. Swipe it themselves. Sign (electronically or on paper). Or enter a PIN number. Write out a check. Grab the right amount of cash. Putting away change. Saving receipts. Entering transaction in checkbook register.

Tracking and understanding their spending. (balancing check books, checking via bank statements, excel spreadsheets, 3rd party apps like mint.)

Returning things, getting credit or cash back.

What tasks are desired?

What they are already doing, with the added rationale from tracking and understanding their spending, the added emotional benefit of rewards involved in decision making.

I want to be able to set up categories. Every purchase I make should end up in a category.

I want it to be able to handle difficult situations like company reimbursements, one time purchases, etc.

I want high level feedback on various categories. Some scale or graphic that is *useful*; with a simple glance I want to know where I am at with my monthly spending on those categories. Am I spending too much or skimping too much?

I want to be able to set up savings categories. At the end of each period I want to see progress and feel good about what my savings was, and maybe what it is going towards.

I want positive reinforcement, I don't want to feel guilty.

Any data entry should be 3 clicks max. 2 clicks is better. 1 click is much better. 0 clicks is the best.

I don't want to have to enter in receipts, numbers, dollars and cents, etc.

One time setup. Automatic.

Payments should be secure (online).

Payments should maximize rewards / reduce cost of item e.g. gas payments with cash is often cheaper at the pump, but rewards on card could outweigh the difference.

Smart. Help prevent making bad purchase before purchase is made.

Encourage better spending habits, as well as when I make good decisions.

Discourage bad decisions while not making me feel guilty.

Understand spending in relation to goals prior to making a purchase for categories where I spend too much money.

Returns to be easy even if I don't remember what card was used or receipt is lost.

Wish that I could understand category spending in different ways for different categories. E.g. auto repairs quarterly vs food spending daily.

Find past purchases, understand past purchases. Purchase lookup. Reference purchase by data, source, overspending.

Understand patterns in overspending - e.g. stay in bed saturday morning until noon and you'll save \$30.

Use bean points as rewards. Create meaning. e.g. gift cards. Build meaning over time besides just feeling good.

How are the tasks learned?

Visual cues in the interface should make learning simple. Worst case, step by step instruction should be made available. But this is not the preferred technique.

Where are the tasks performed?

Anywhere a business transaction is performed. E.g. At businesses, online, and over the phone.

What's the relationship between customer & data?

The customer should be required to provide as little input as possible during the process. Data is analyzed and provided at appropriate times to the customer automatically and available to the customer on demand.

What other tools does the customer have?

Traditional payment methods, cash, check books, everything they have now. Bank statements, 3rd party apps like mint.

How do customers communicate with each other?

They don't. There could be potential for integration of social aspects and milestones.

How often are the tasks performed?

Every day.

What are the time constraints on the tasks?

Less than 10 seconds for simple tasks, less than 30 seconds for moderate tasks, less than 2 minutes for difficult tasks.

What happens when things go wrong?

The user can always undo / go back / find an exit. Interface is always visual cuing for avoiding mistakes.

Task Descriptions

There are several subtasks people perform when purchasing/paying, which played into our first two tasks:

1. Decide whether or not to purchase (feelings + rationale)
2. Select a payment type (cash, credit card, check, points, alternatives)
3. Give payment to merchant
4. Deal with transaction details (change, tip, signing CC slips etc.)
5. Organization of payment artifacts (save receipt, enter transaction into checkbook, put away change, put away CC, etc.)

Task 1, buy a latte (simple):

You walk past a Starbucks and decide you want a coffee. You recently got into the fall spirit, so you order a grande pumpkin spice latte. Pay for your drink.

This is the main, everyday abstract payment task that is the majority of usage of any payment system. It should be, fast, simple, and error-free.

Task 2, buy a little (medium):

You are a financially minded small business owner. Today is cyber Monday and you've found a nice TV and DVD player online that you want to buy for your waiting room. You've got them in your cart and are ready to check out. Do so, ensuring that you get the most reward points for your purchase AND ensure you've got what you need to document the business purchase.

The world of payments online is a little more complex and so is ensuring you've got your receipts saved and your expenses separated out. This might not be easy, but it shouldn't be a big challenge either.

Task 3, examine your spending and update your budget:

You decide you want to have a better understanding of your spending and save a little more each month, so you decide to get down to work. Find your largest three categories of spending on a month to month basis that isn't housing, transportation, utilities, or groceries. Then choose one of those categories and create a budget for that category so you can save more than you are currently saving now, and find a way to hold yourself to that budget for the next month.

Initial Sketches and Storyboards

We came up with three initial designs...

Design #1

In design #1 we started sketching basic ideas based on our contextual inquiry. Our first thought was that we wanted to be able to review basic financial data on the landing page and then drill down into our various spending categories.

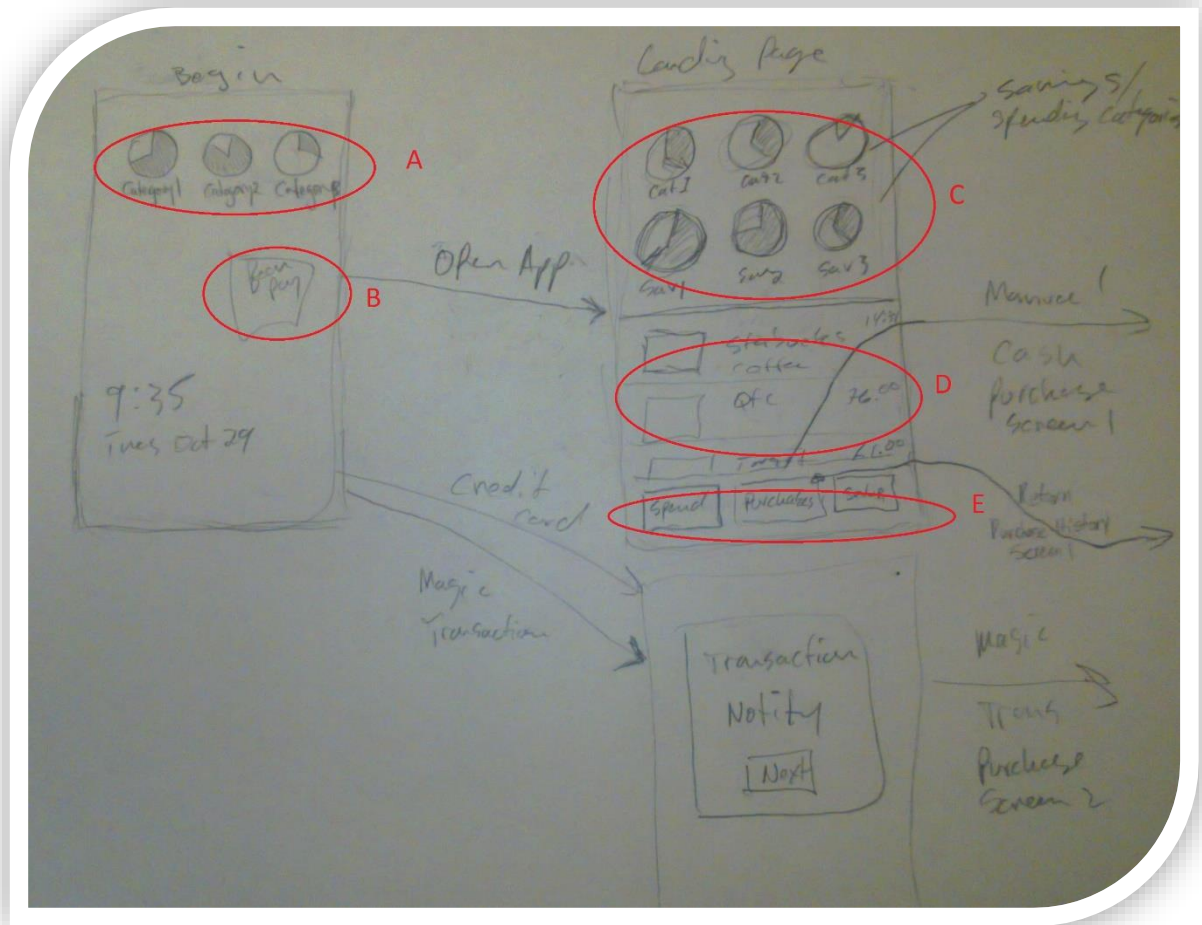


Figure 1

In figure #1.A we see our spending categories on the main page. There is also a button to extended detail. In figure 1.C-E we have extended categories for saving/spending as well as timeline of recent purchases.

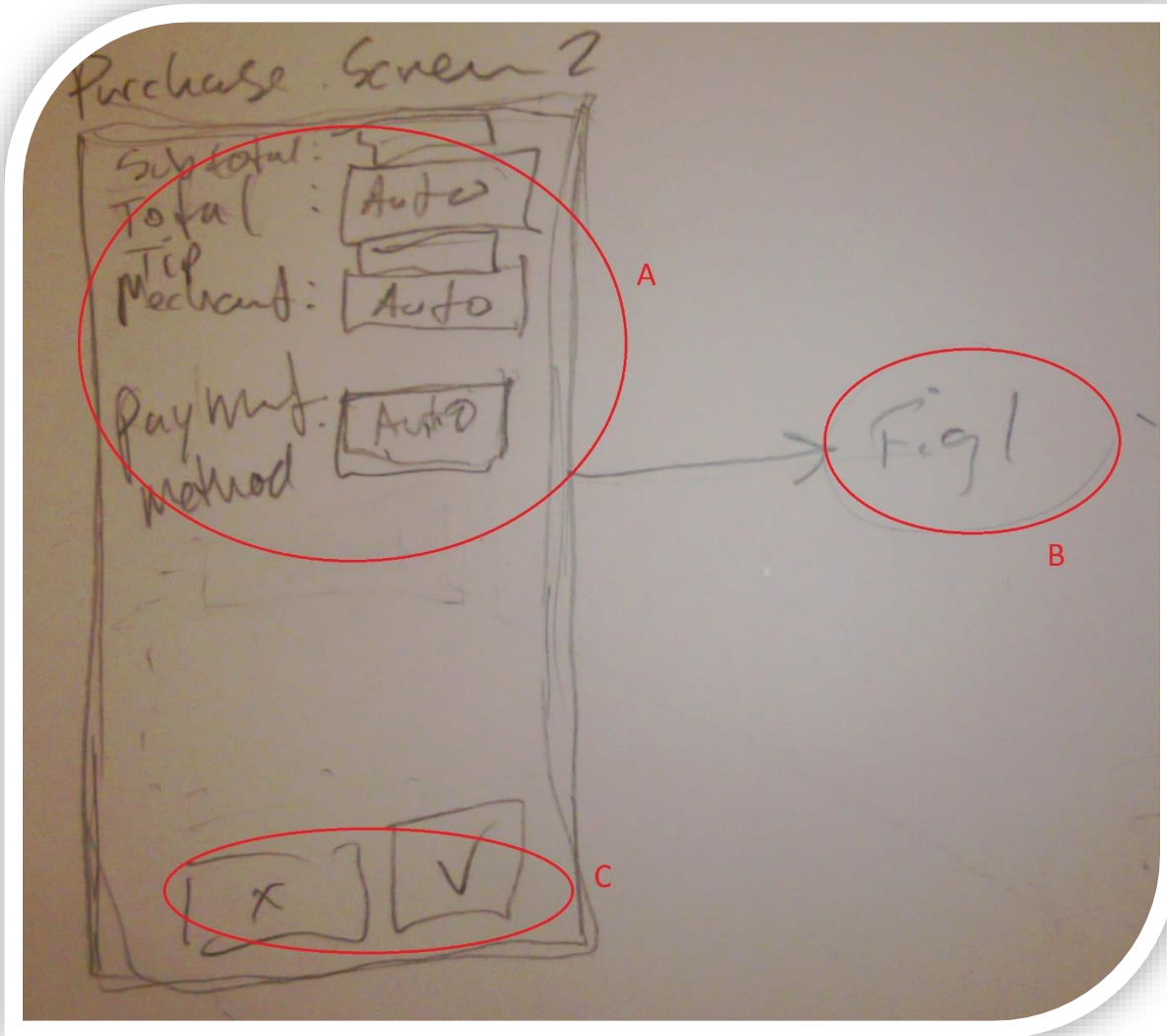


Figure 2

In figure #2 we see a purchase confirmation screen. This has all fields auto-populated with payment amount, merchant etc.

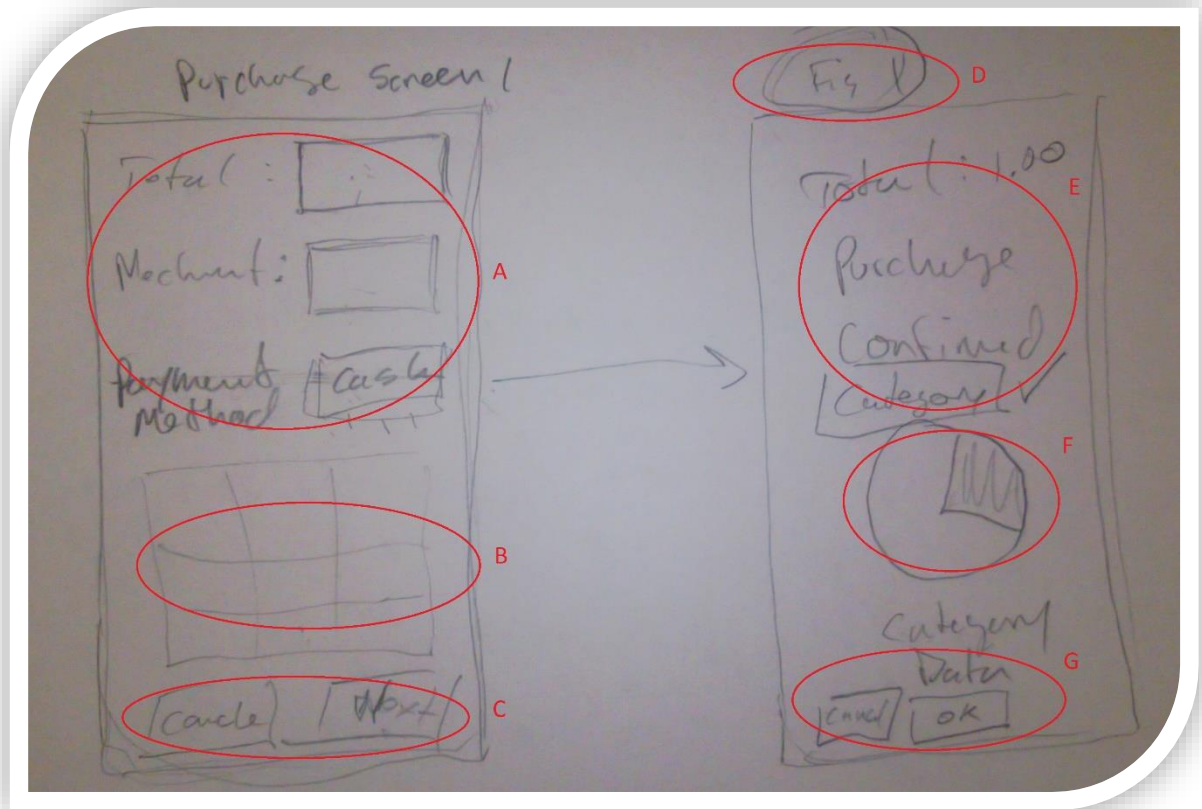


Figure 3

In figure 3.A-C we see the purchase screen from figure #2 without the fields auto-populated. This gives the user the ability to track cash purchases. Figure #3D-G show the purchase confirmation page and how it tracks against the user's budget for the particular category.

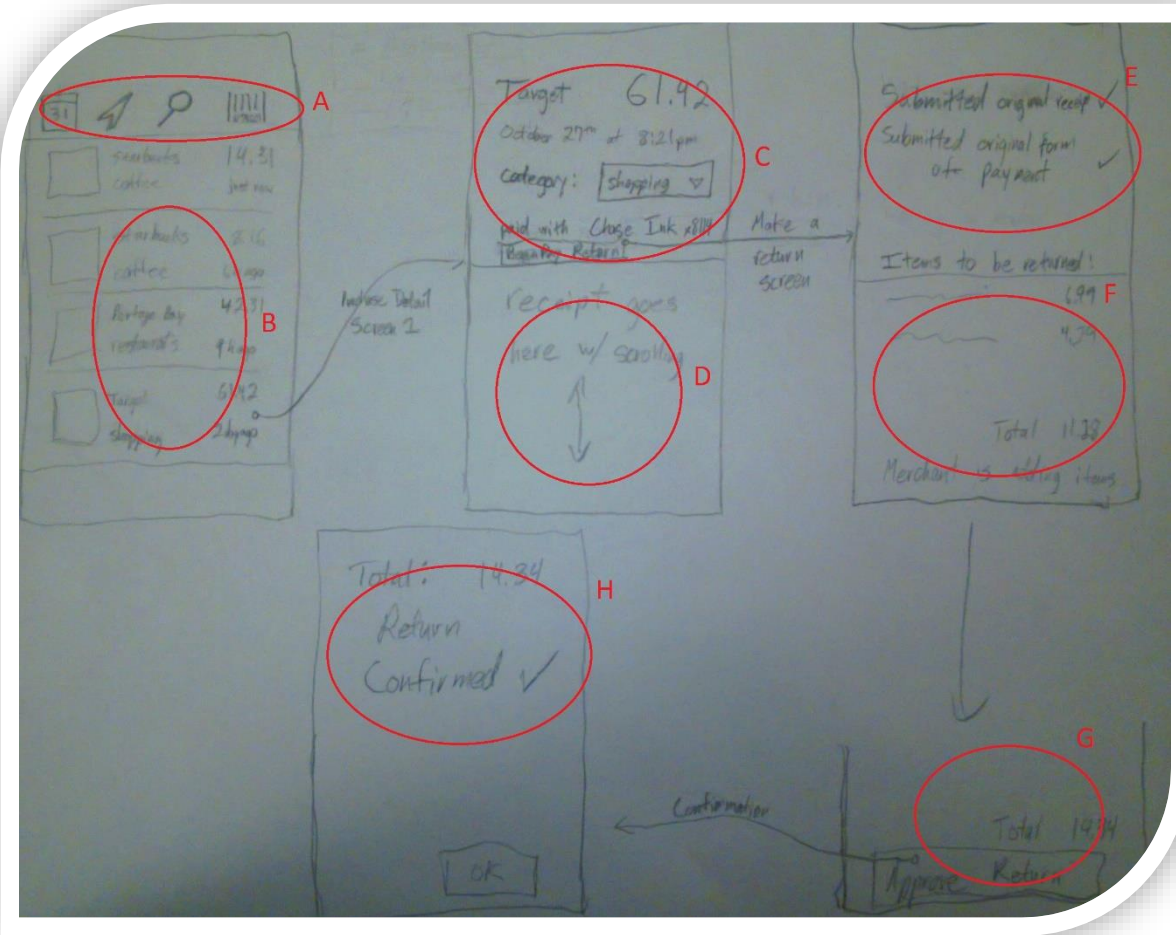


Figure 4

If the user drills down into the purchase history he will see a timeline of purchases. By selecting a purchase the user can get an electronic receipt and even link with merchant payment system for returns.

Design #2

In Design #2 we took the approach that the user would either make the payment from within the app or immediately be notified of a new transaction you make using an existing credit card. Notice in figure 5.A – a notification notifying the user of a new transaction.

When the user clicks on the transaction he/she is immediately taken to a screen that they can select a category (Fig 5.C) if one can't be determined by machine learning.

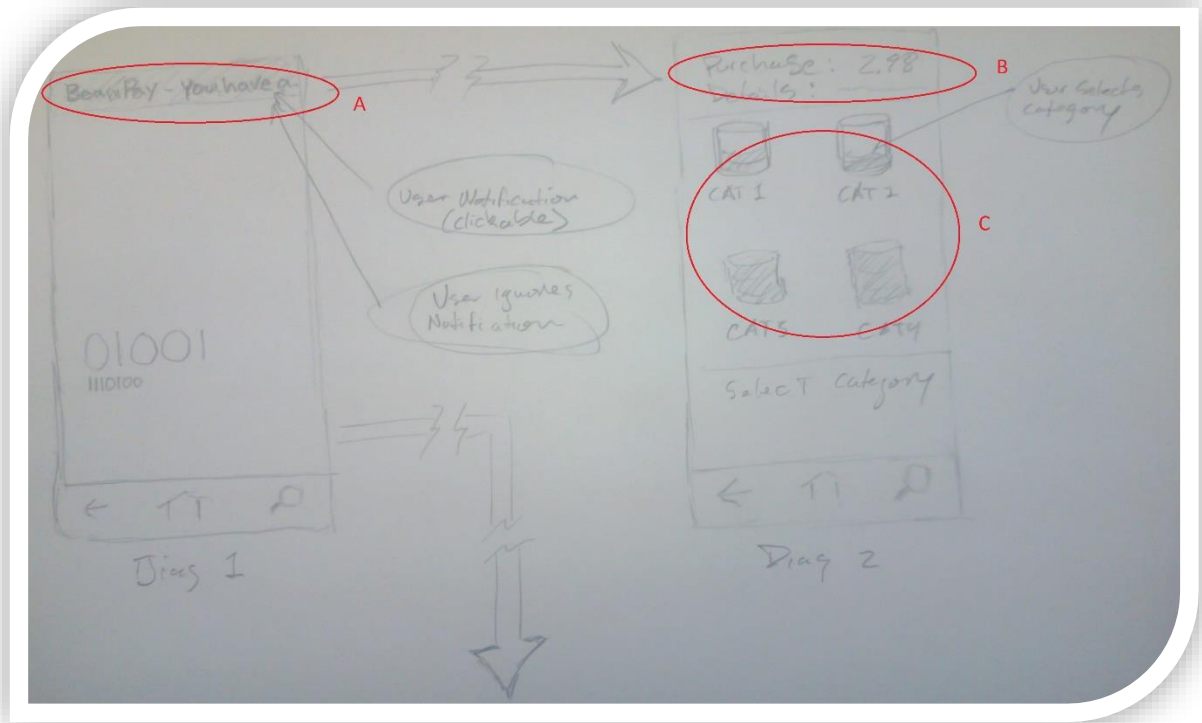


Figure 5

In figure 6.A we see a list of the most recent transactions that need user attention. By selecting the '+' icon the transaction is added to the "Spend" categories. If the user clicks on the '-' icon the transaction is removed from the feed. This would happen if the purchase was reimbursable etc. Figure 6.C-E lists the users spending budget as well as savings.

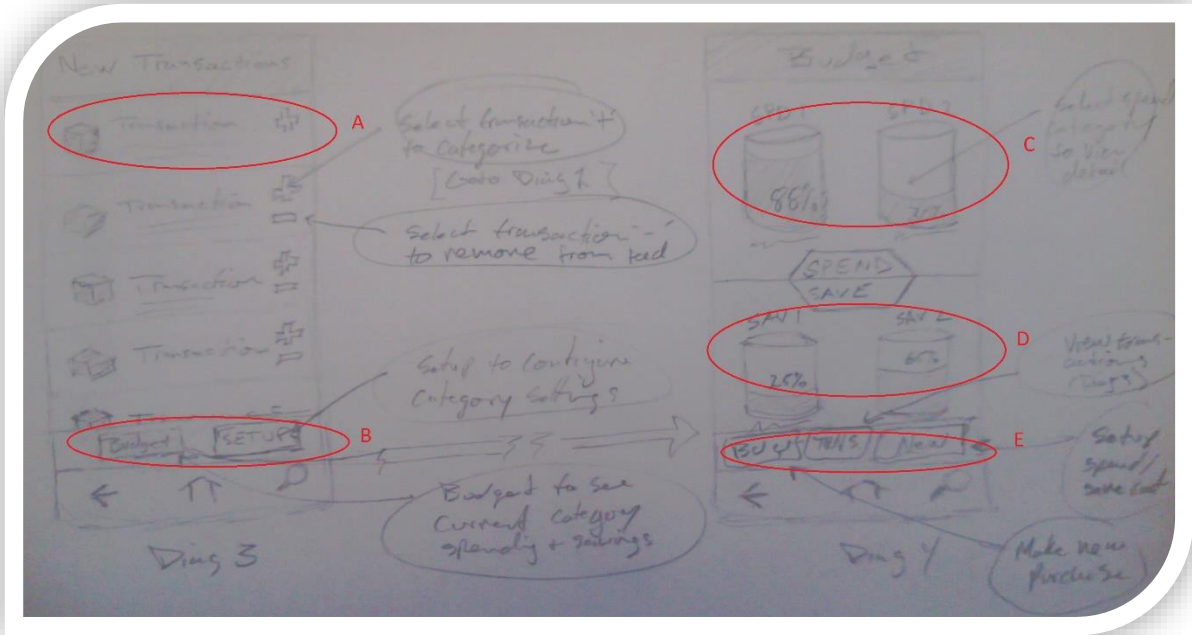


Figure 6

If a user clicks on a particular budget icon in figure 6 it will take him to Figure 7.A. Here the user can see all the recent transactions related to this category as well as historical data and settings.

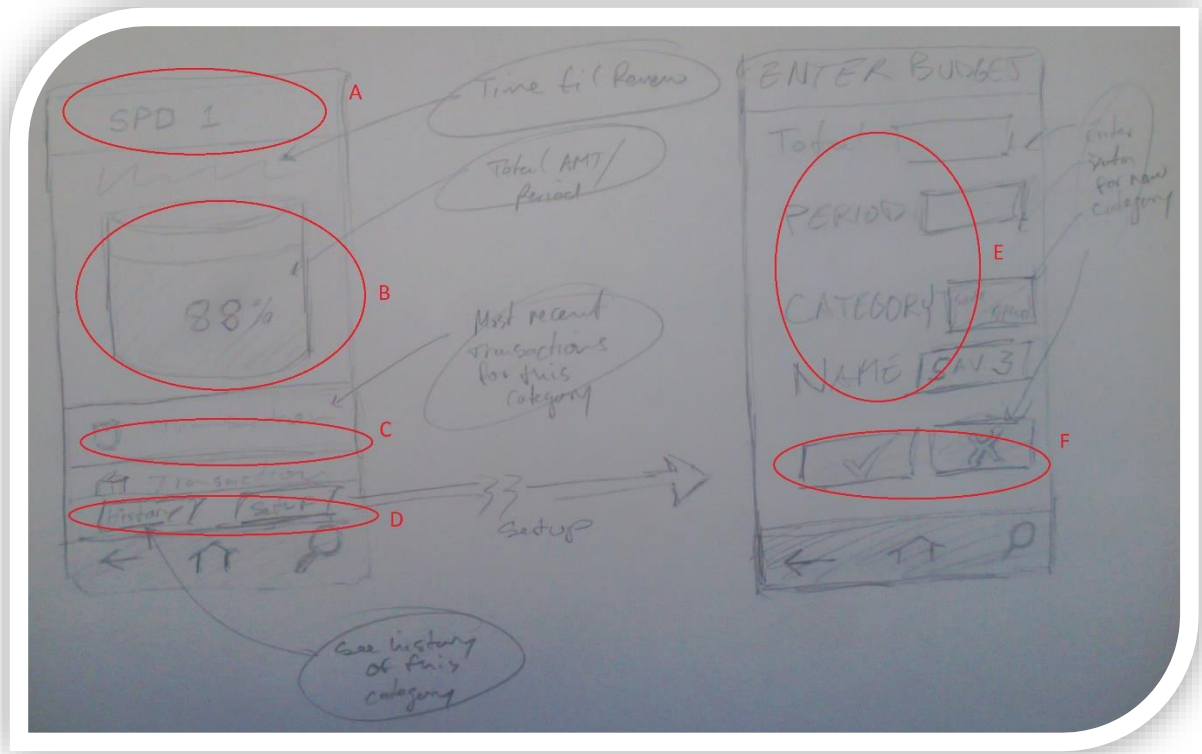


Figure 7

Figure 7A-D shows a particular budget category and Figure 7E-F shows how a user would setup a particular budget.

Design #3

In design #3 we wanted to focus on the payment side of the interface – exploring how various payment sources would be managed.



Figure 8

After opening up the BeanPay interface (Figure #8), it displays the MagicPay requested transaction, monthly savings information, a suggestion on whether or not to make the purchase, the suggested form of payment to maximize rewards, and present the user with options to accept or decline to pay.

If you decline the transaction, it verifies that you want to do so. Swiping to go back takes you back to the previous screen. Swiping forward to cancel the transaction does so, and then shows a transaction canceled confirmation screen.

If, on the original purchase screen, you tap on the suggested payment method, it provides you with all your payment options. You can select an account by tapping on it, and then swipe right on the "ready to pay" ribbon.

Once you approve the transaction, you are taken to one of two screens. If it is a service establishment, you choose to select a tip level, which when completed would take you to the final confirmation screen. If you are at a non service establishment, it takes you to the final confirmation screen. On this screen, you have the option of adding tags to the purchase and can see your updated expected savings for the month.

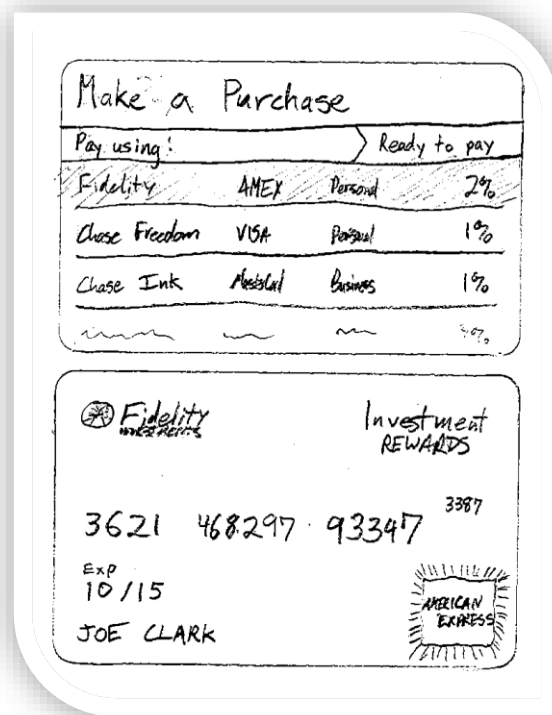


Figure 9

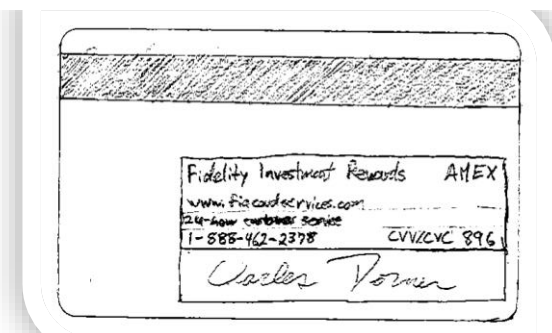


Figure 10

Light/pressure sensors in the BeanPay interface detect that it has been removed and is in your hand and it turns on. It immediately opens the make a purchase screen, allowing you to select your payment account by tapping on it and swiping across the 'ready to pay' ribbon. The BeanPay interface then transforms its display to appear like a normal credit card, the credit card contact information and CVV on the smaller display screen on the back, and is now ready to transmit payment information to the merchant.

Selected interface design

During the contextual inquiry process and design sketching we focused on several themes and problems. On the one side, there is the problem of user payments – ability to manage and keep track of all payment sources. On the other, keeping track of a budget a managing spending categories.

In our first prototype we explored various methodologies for payment tracking as well as budgeting. One component of this system that we identified as being useful was a payment timeline which tracks payment history. From here the user can quickly categorize and analyze a payment.

In our second prototype we focused more on the budget side – giving the user control of setting up categories for spending and tracking them against current budgets.

In our 3rd prototype we explored an interface for payments and a credit card replacement. During our inquiry it became clear that while people might use a credit card form payment device for paying, they want a more powerful platform for performing other tasks – for example they expect notifications to happen on their mobile computing device. They want to be able to dive deep into purchase information on their desktop computers. They expect to be able to view, understand, and examine their financial data on any computing platform.

Thus we concluded that given today's technology a multi-device system would be necessary – a single store of information online, with multiple interfaces into that system from a variety of devices. However, since we are not working within the constraints of today's technology we believe that we can design a single interface around a screen with sufficient compute power and connectivity to transmit payment information and track a budget. The BeanPay interface allows for standard CC purchases and BeanPay Magic payments, manual entry for non-credit card purchases, provides notifications to support saving money and categorization of non-BeanPay transactions and the basic exploration of spending and saving data.

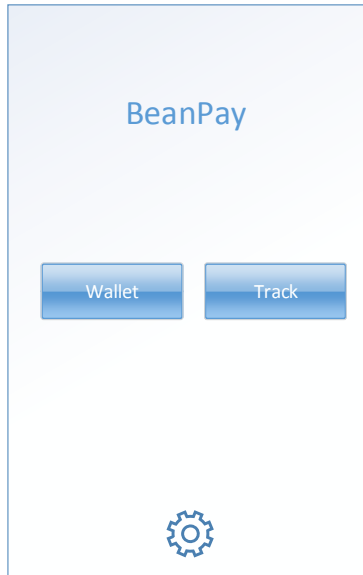


Figure 11

In figure 11 we have combined the wallet aspects of our interface with the budgeting aspects. From this main landing page we can select either wallet to make payments or budget to track transactions and see budget. Clicking the wallet button takes the user to figure #12.

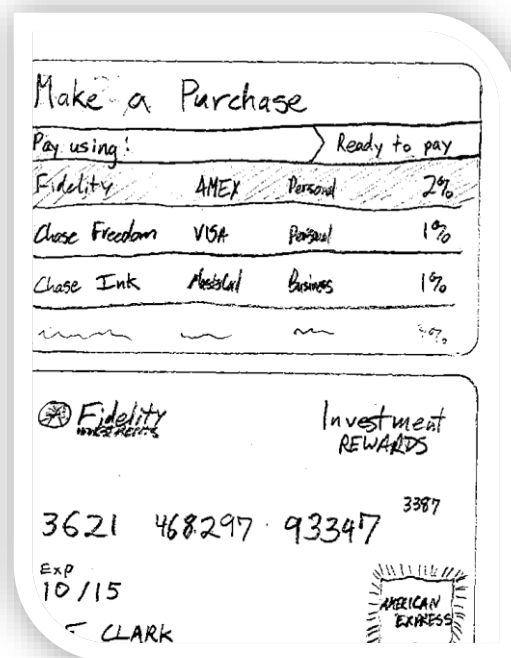


Figure 12

Clicking on the track button icon takes the user to the transaction timeline along with budget categories. Clicking on the gear takes the user to setup menu to either add payment sources or to setup a budget.



Figure 13

Figure #12 shows the basic layout for the “Track” page. From here the user is presented with a list of the most recent transactions. The +/- icons indicate that a user can remove the transaction from budget timeline or add to a budget category. If the user selects the budget icon they will be taken to Figure 6C-E.

BeanPay Features
1 – Manage Payment sources – e.g. Credit cards, cash.
2 – Process payments.
3 – Track payment history.
4 – Setup budget categories.
5 – Notify users of transactions.
6 – Intelligent category detection.
7 – Manual category selection.
8 – Present historical data for budget monitoring

Our final design will provide the above primary features. We will combine elements from all three of our sketches to provide a one-stop payment and budgeting interface.

Scenarios

Scenario 1A, buy a latte (standard):

Pull the BeanPay card out of your wallet. After it powers up, select the credit card you want to pay with by swiping to the right on the row that describes the card you want to pay with. Hand the card to the cashier, who swipes it. Take back your card and your receipt.

Scenario 2, buy a little (medium):

On your desktop web browser, click the BeanPay option. Your wallet vibrates. You open your wallet and the BeanPay card is already awake, so you take it out of your wallet. It is displaying the payment request and showing you that the payment option is your 2% cash back rewards card. You slide your thumb over the approve ribbon. The payment is approved and the receipt displays on your card, as well as tagging options. You swipe across the business tag to select it. The website updates on your screen acknowledging the purchase.

Scenario 3, examine your spending and update your budget

You pull out your mobile device and open BeanPay. You select “Track” on the main screen and seeing various Spending and Savings categories on your device. You swipe through them, noticing they are sorted by how much you spend on each area. After Rent, Food seems to be a big spending category, as does Entertainment. You tap on the food category and see more details – you only spend about \$170 a month on groceries, but almost \$400 a month on eating out. You swipe through the transactions and see that you are eating out every single day, and sometimes twice a day. You decide that you could spend less money on eating out and save a little more there. You tap the “change budget” button and drag the slider left until you hit \$300 / month. You save the budget and look forward to saving over the next month.