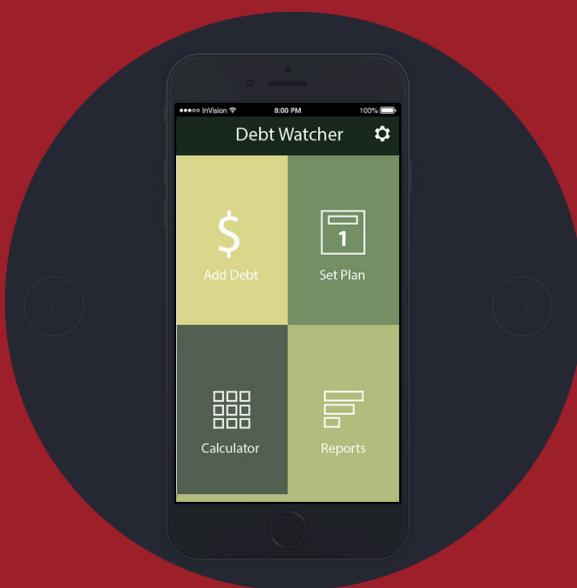


Contents

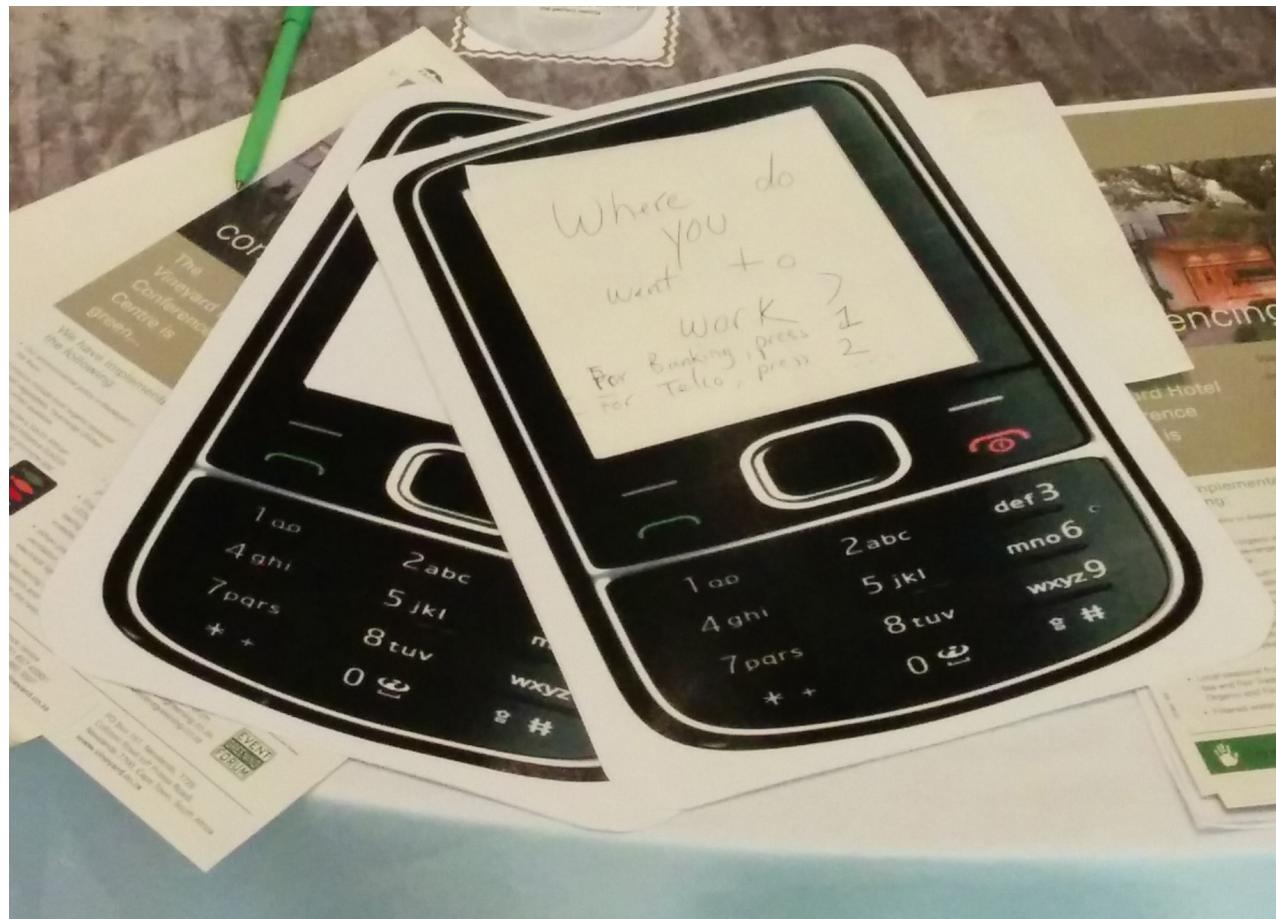
● App Project

Introduction	1
Research	2
Prototype & Test	3
Wire Framing	4
Final App	5

App Project



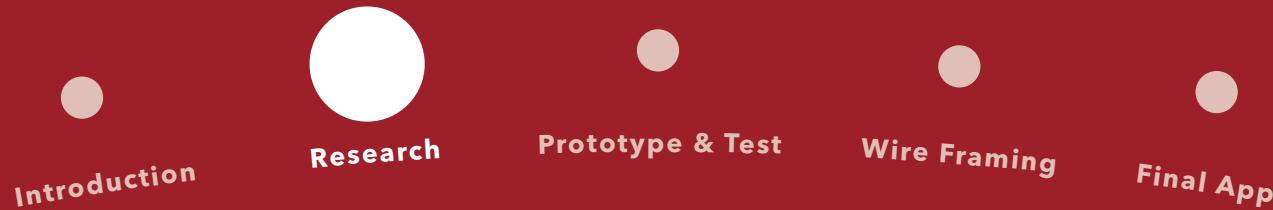
Calculators | Introduction



Calculator App Project.

In this project I will design a calculator app of my choice that gives the user great experience, its functions are well understand by the user, and is aesthetically beautiful. I will be following the process of brainstorming ideas, defining an audience, sketching and building prototypes, and user testing them. Based on the user's observations, I will implement the improvements to my app.

App Project



Calculators | Research

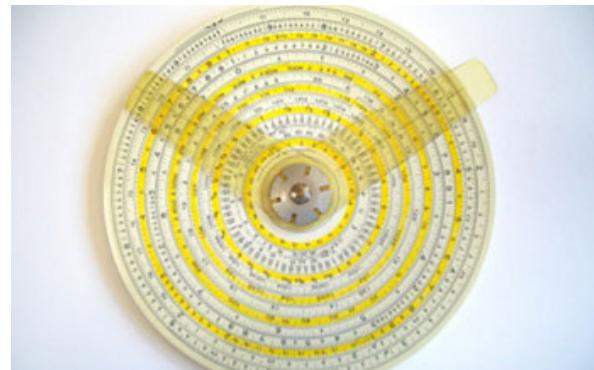
Abacus.

The abacus, a sort of hand operated mechanical calculator using beads on rods, first used by Sumerians and Egyptians around 2000 BC.



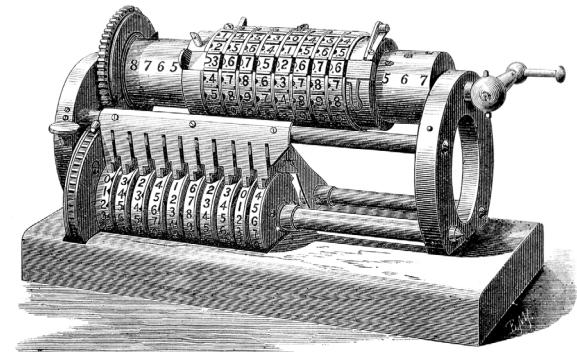
Slide Rule.

The slide rule is basically a sliding stick (or discs) that uses logarithmic scales to allow rapid multiplication and division.

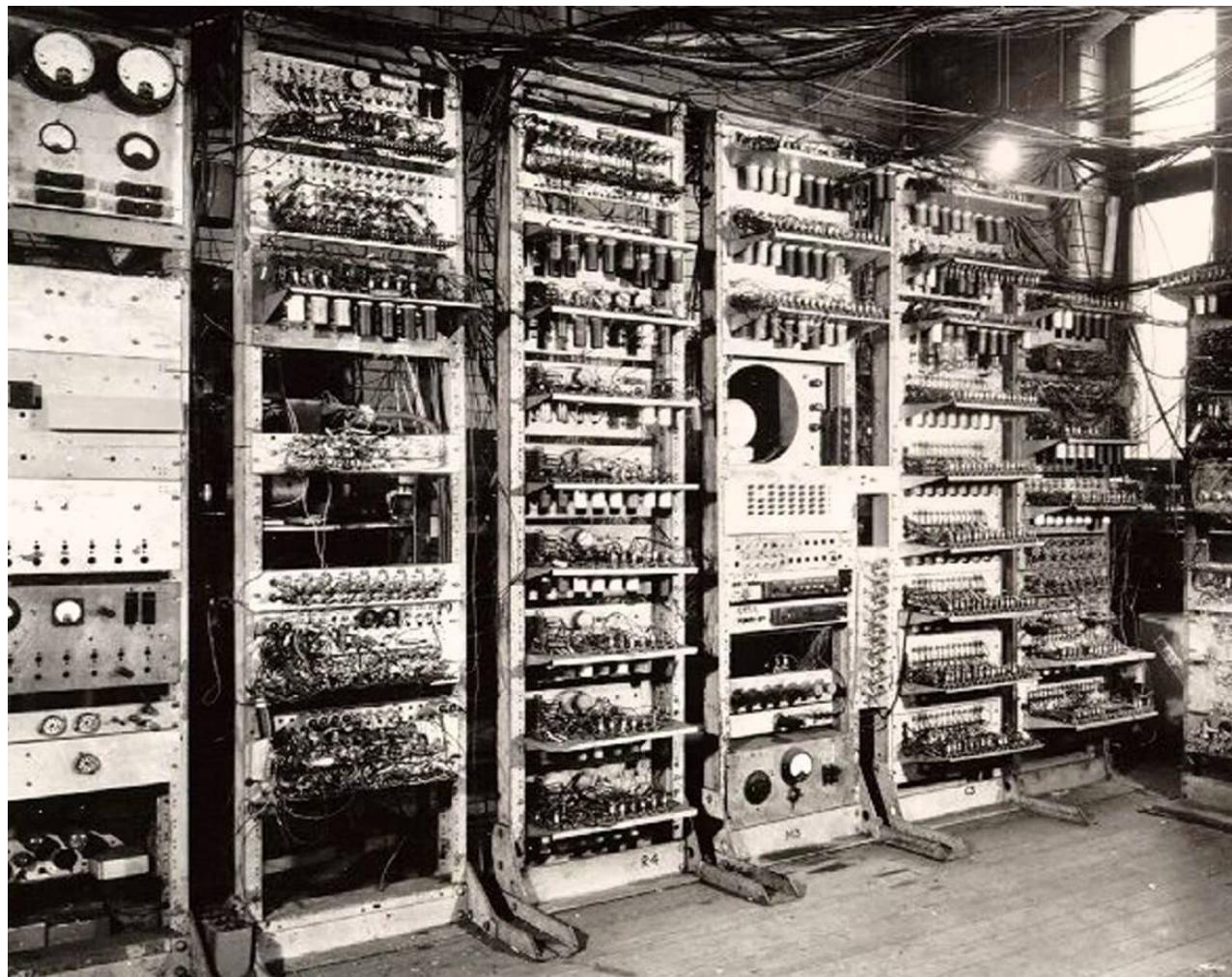


First Mechanical Calculator.

The first mechanical calculator appeared in 1642, the creation of French intellectual and mathematics whizz kid Blaise Pascal as "a device that will eventually perform all four arithmetic operations without relying on human intelligence."



Calculators | Research



Electronic Calculator.

The first solid state electronic calculator was created in the 1960s, building on the extensive history of tools such as the abacus, developed around 2000 BC, and the mechanical calculator, developed in the 17th century. It was developed in parallel with the analog computers of the day. In 1946 people built ENIAC.

ENIAC was 1,000 times faster than electro-mechanical computers and could hold a ten-digit decimal number in memory. But to do this required 17,468 vacuum tubes, 7,200 crystal diodes, 1,500 relays, 70,000 resistors, 10,000 capacitors and around 5 million hand-soldered joints. It weighed around 27 tons, took up 1800 square feet of floor space and consumed as much power as a small town. Not exactly a desktop solution.

Smart Phones | Research

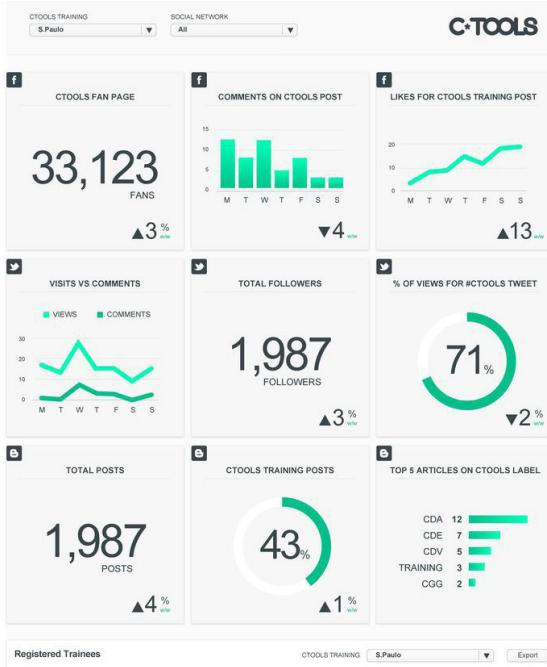


Technology and interactions that Smart Phones afford

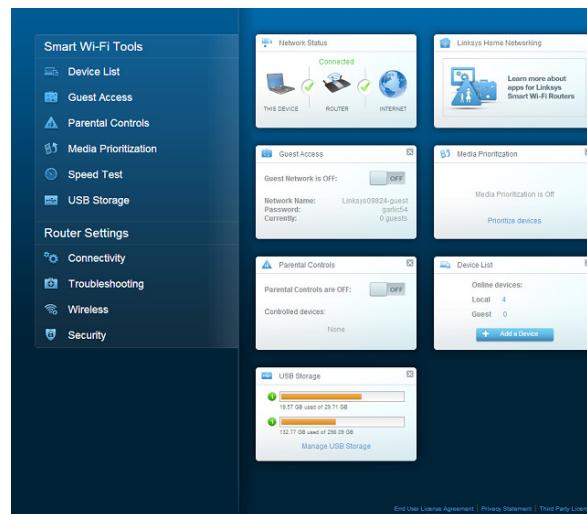
- Phone calls
- Text msg
- Email
- Video calls
- Weather service
- Interactive maps
- Video streaming
- Photo msg
- Music listening
- Media downloads
- Web browsing
- Online shopping
- Online services
- Calendar
- Planner
- Note taking
- Voice recording
- Books reading
- Electronic games playing
- Social media access
- Photographs
- Video
- Communication platforms
- Cloud service
- Calculators

Interfaces | Research

Nice hierarchy and feeling of unity.



Clean navigation design.

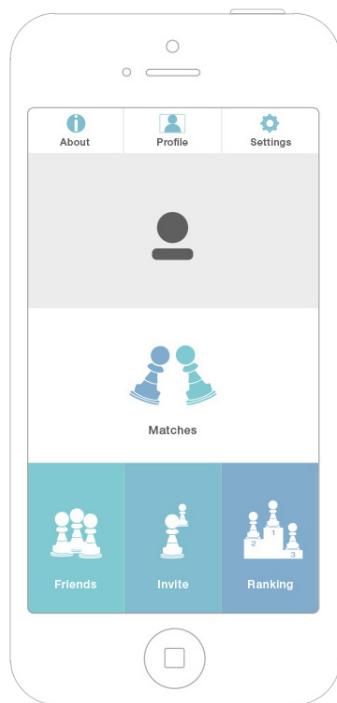


Modular and uniform design of the icons.

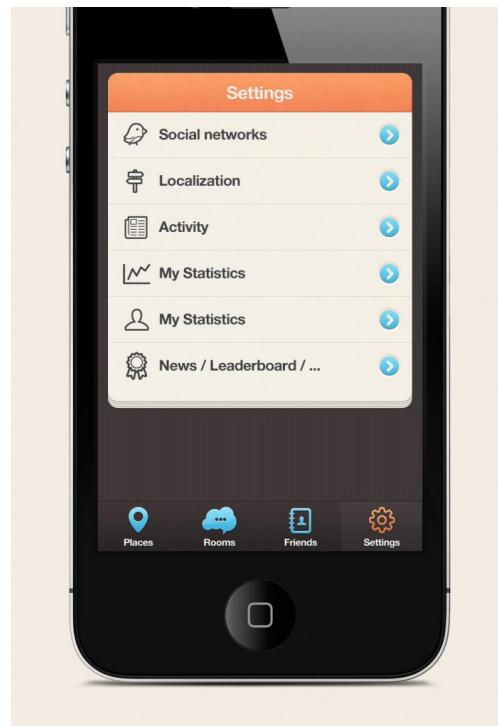


Interfaces | Research

Simple design makes it easy to understand.



Colors used to indicate where the user is.



Strategic use of colors to facilitate navigation.



Exploration | Research



List of Possible Calculator's Apps

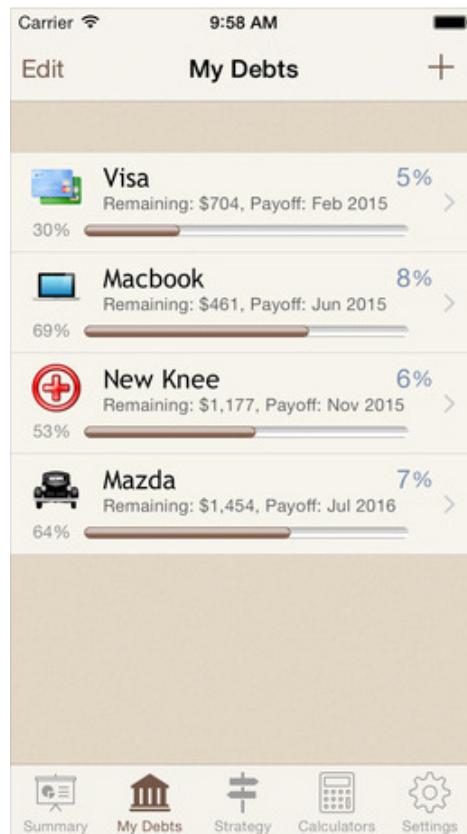
- Loan calculator
- Pregnacy calculator
- Tips calculator
- Weight calculator
- Bill splitter
- Unit conversion
- Trip cost calculator
- DUI calculator
- Debt watcher
- Cars' price comparison
- Baby sitter calculator

My choice for the App Project

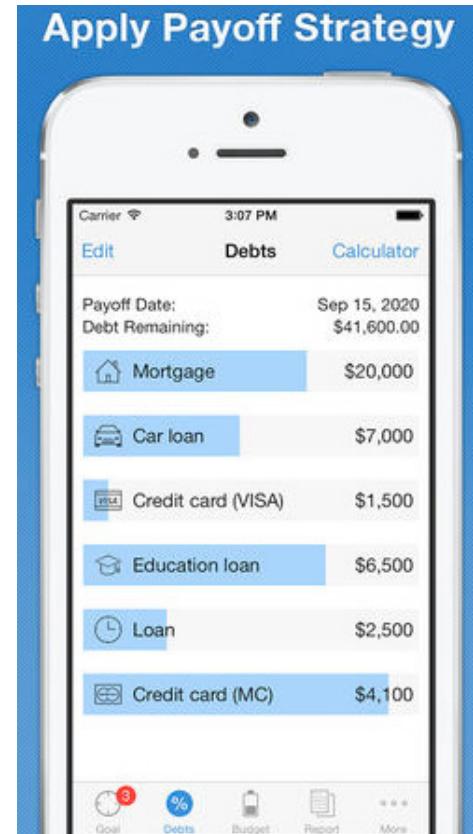
- Debt Watcher

Calculator Competition | Research

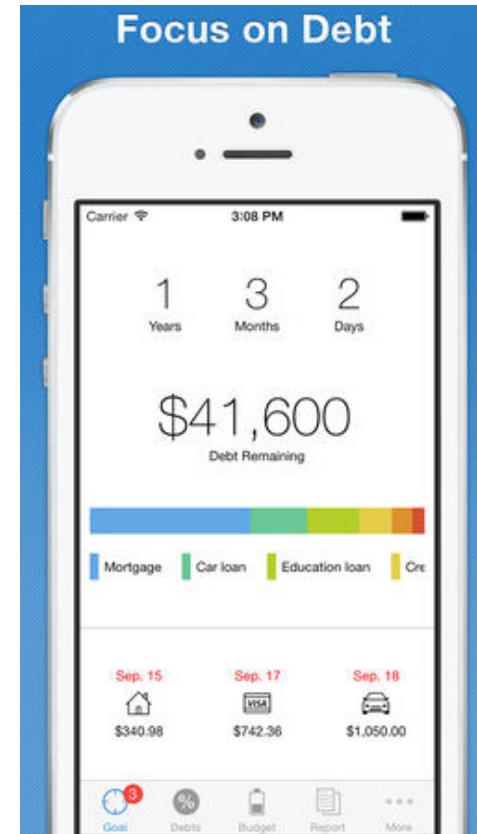
I like the icons that helps to recognize the type of debt, but the information is kind of complicated to read at first glance.



I like simple way to show data here, but it's unclear whether blue bars represent part of the total debt or individual amounts.

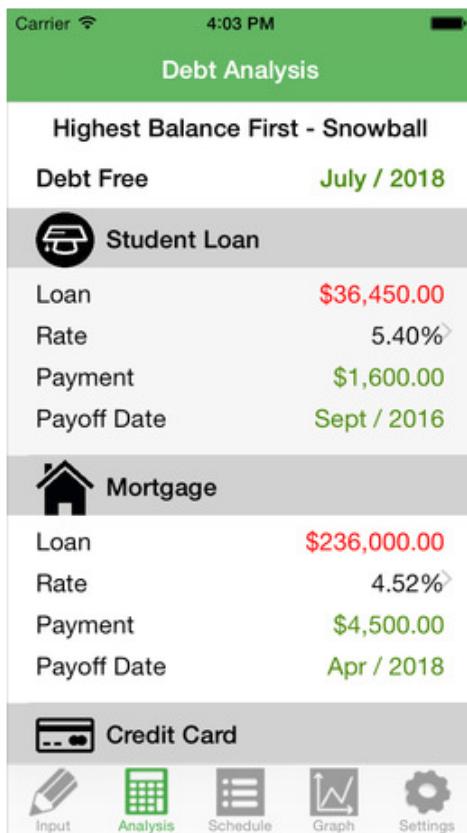


This screen of the same app is a bit too busy and there is no sufficient grouping. Good thing is that design continue simple.



Calculator Competition | Research

I like the use of colors to establish hierarchy and simplicity of the design.



This screen looks too busy because has too much information for this kind of device. They could use better hierarchy, like on previous screen.

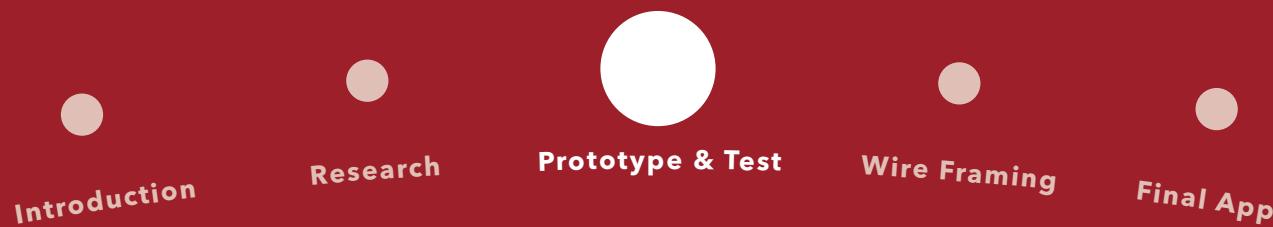
Debt Schedule				
Total Debt		\$329,803.00		
Total Interest		\$34,174.57		
Total Repaid		\$363,977.57		
Date	Payment	Interest	Principal	Balance
10/2016	7900.00	625.14	7274.86	1443/2.46
11/2016	7900.00	596.59	7303.41	137069.05
12/2016	7900.00	567.92	7332.08	129736.97
1/2017	7900.00	539.14	7360.86	122376.11
2/2017	7900.00	510.24	7389.76	114986.35
3/2017	7900.00	481.23	7418.77	107567.58
4/2017	7900.00	452.10	7447.90	100119.68
5/2017	7900.00	422.85	7477.15	92642.53
6/2017	7900.00	393.49	7506.51	85136.02
7/2017	7900.00	364.01	7535.99	77600.04

Input **Analysis** **Schedule** **Graph** **Settings**

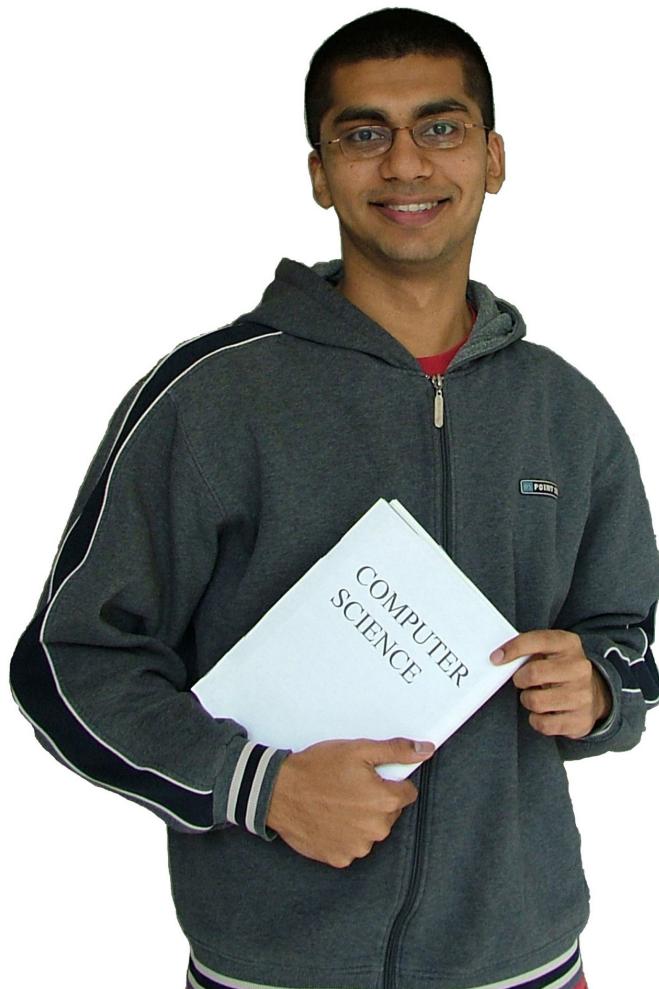
In this screen I found an attempt to use colors to address the hierarchy, but turned to be very busy and difficult to read.

Credit Card	
RESULTS	
TOTAL YOU WILL PAY	21,516 USD
INCLUDING INTEREST OF	16,516 USD
MONTHS TO REPAY	318 26 years
AGE REPAYED BY	61 Change age
SAVE MONEY!	
INCREASING PAYMENT TO	74.44 USD >
SAVES YOU	8,818 USD
TOTAL YOU WILL PAY	12,698 USD
INCLUDING INTEREST OF	7,698 USD
MONTHS TO REPAY	171 14 years
AGE REPAYED BY	49 Change age

App Project



Persona | Prototype & Test



Tom Brand.

24 years old, finishing his degree in IT.

Tom is at the beginning of the carrier in decent IT company. After several years of college, he accumulated significant amount of debt which he is determined to pay off now. He needs a way to control amounts and dates of payments. tom is also interested in an app that helps him to set the appropriate strategy to deal with his debt while he will provide for his basic needs.

Scenarios | Prototype & Test

Scenario 1

Your new debt watch app will help you analyze your situation. In order to do so, you need to add all your debts to the system.

Add your debt to the app's data base.
(student loan \$65 k, 3.40% APR, monthly payment \$470.00, during 15 years)

Scenario 2

Your app can display statistics about your debt such as comparing interest amounts for your loans or show how much left to repay the loans.

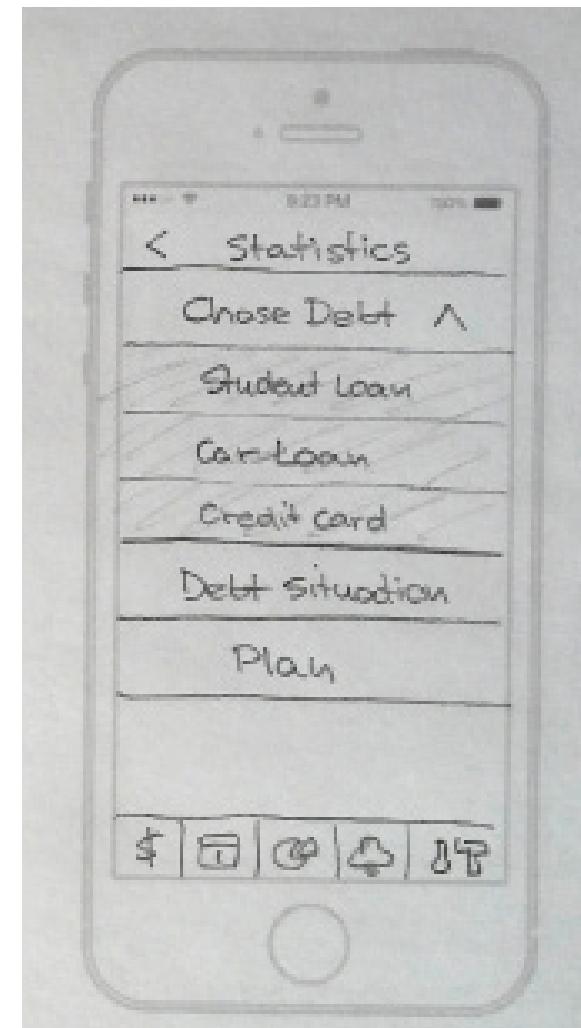
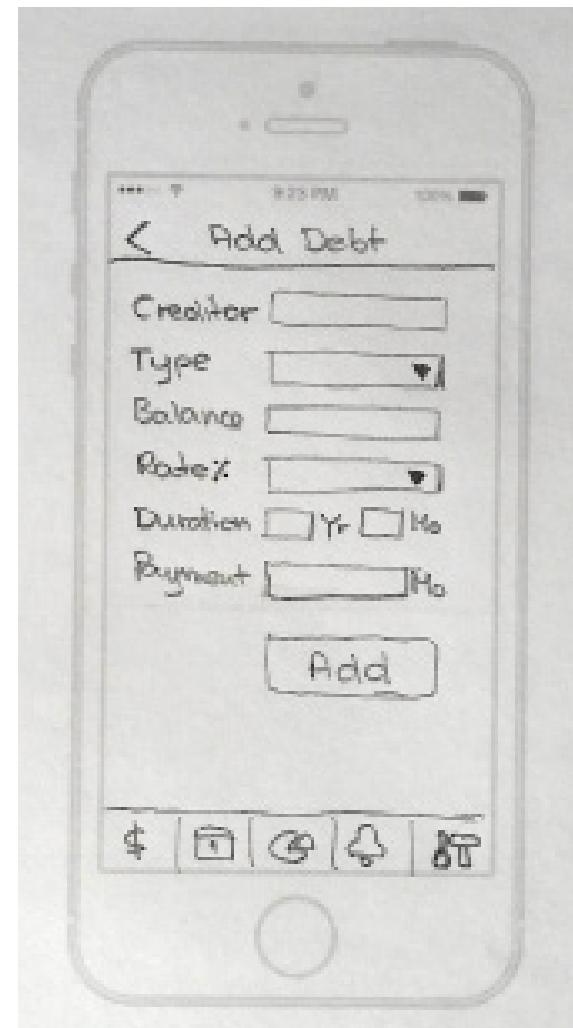
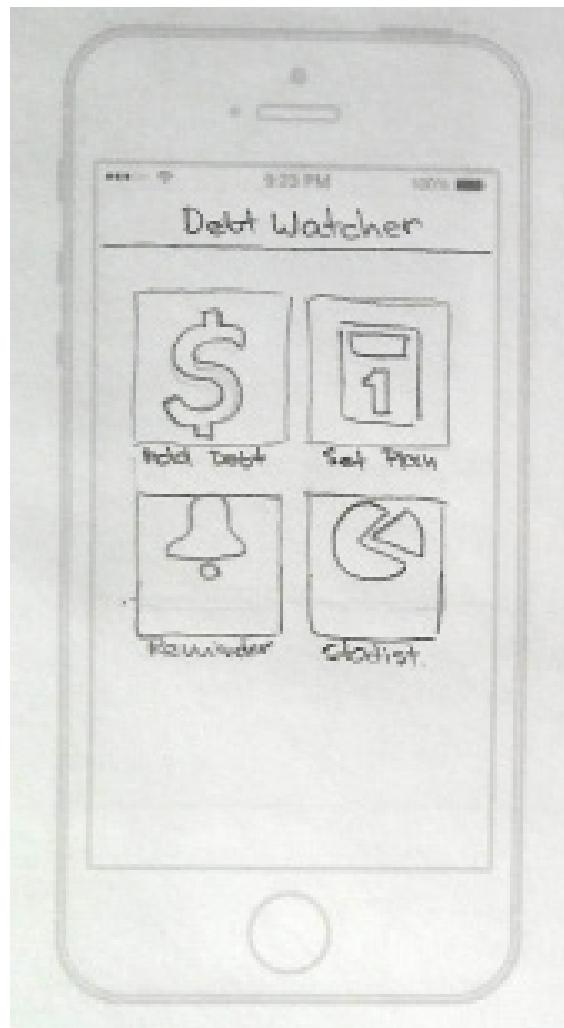
Find the graph that tells you how much of the loan you still need to repay.

Scenario 3

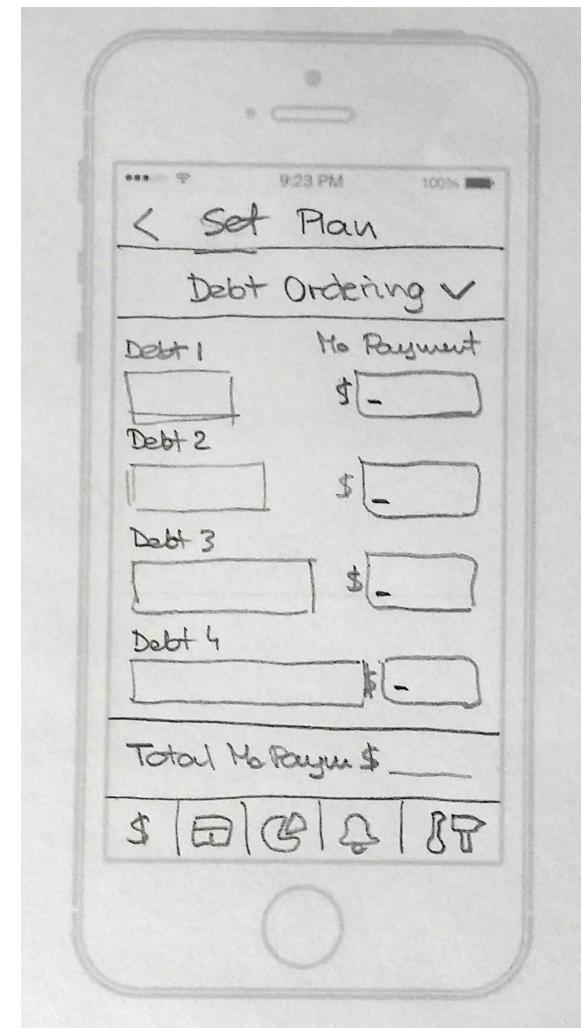
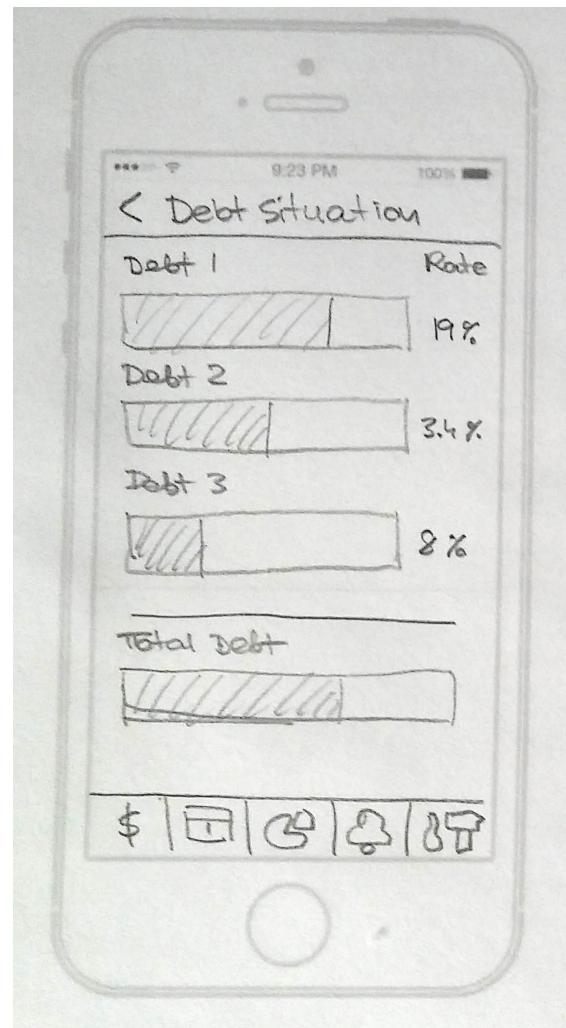
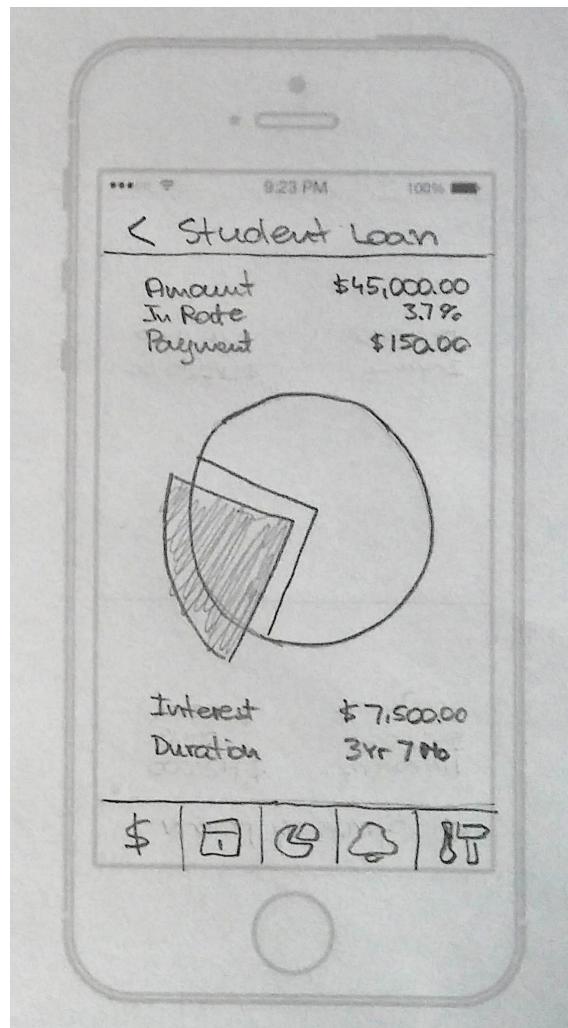
Your app helps you schedule payments and remind you about it.

Set the reminder to not miss the monthly payment on your loan.

Sketches | Prototype & Test

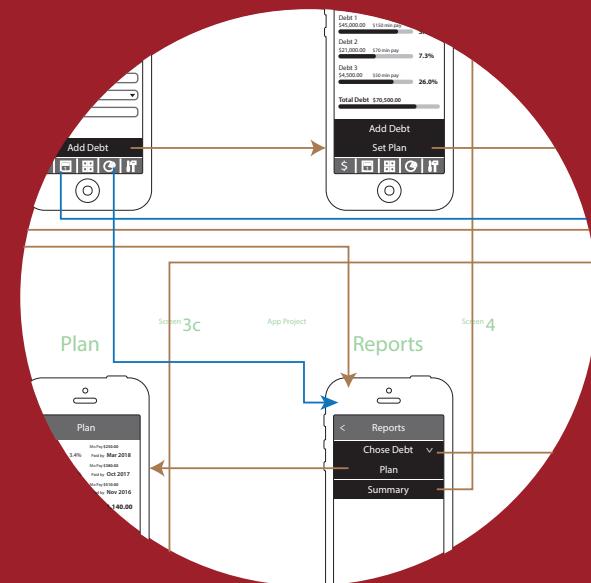


Sketches | Prototype & Test



App Project

Introduction Research Prototype & Test Wire Framing Final App



Scenarios | Wire Framing

Adjusted scenarios

Scenario 1

Your new debt watch app will help you analyze your situation. In order to do so, you need to add all your debts to the system.

Add your debt to the app's data base.
(creditor Student Loan, student loan \$65k, remaining balance \$45k, 3.40% APR, minimum monthly payment \$150.00)

Scenario 2

Your app can display reports about your debt such as comparing interest amounts for your loans or show how much left to repay the loans.

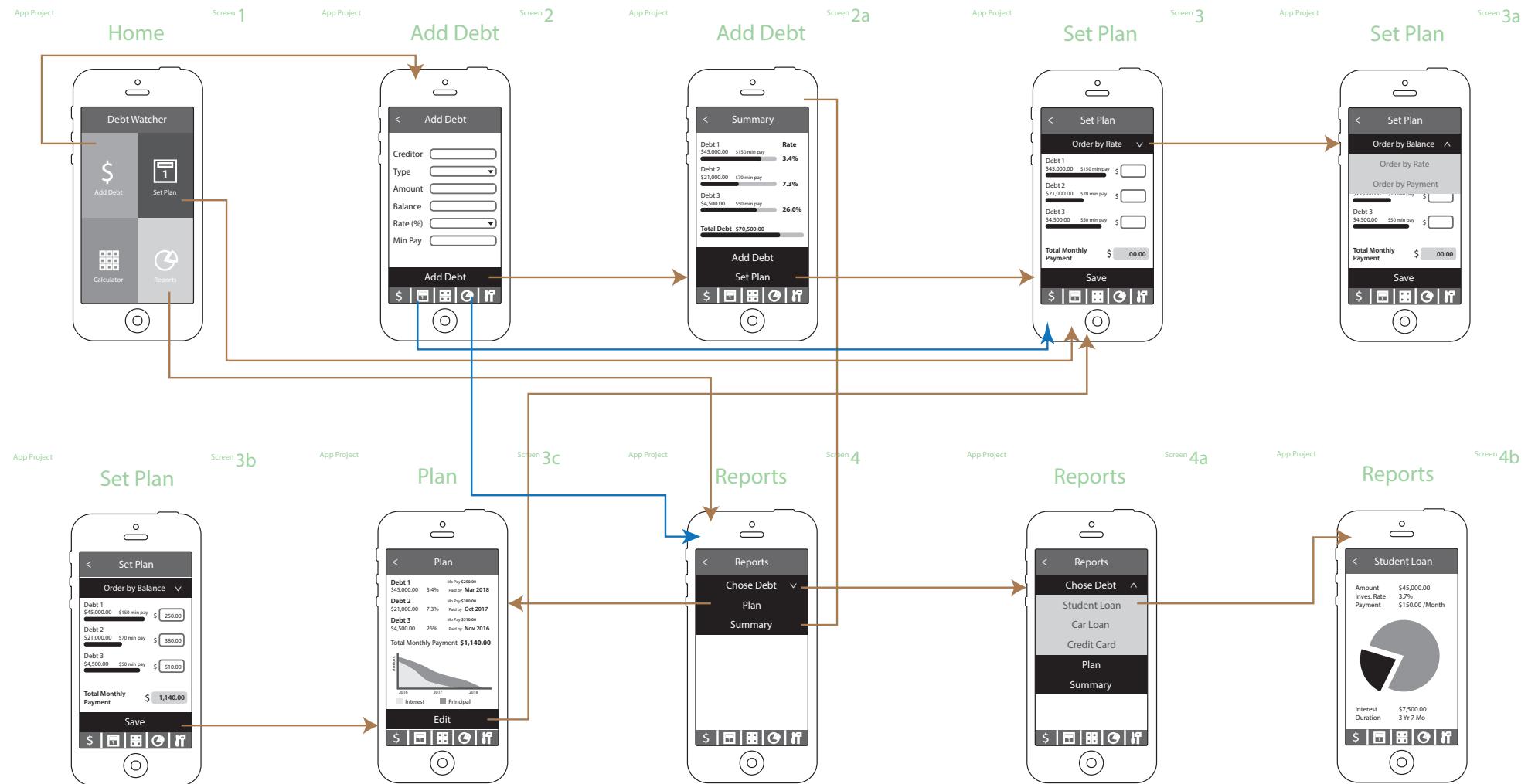
Find the graph that tells you how much of the student loan's interest you will have to repay.

Scenario 3

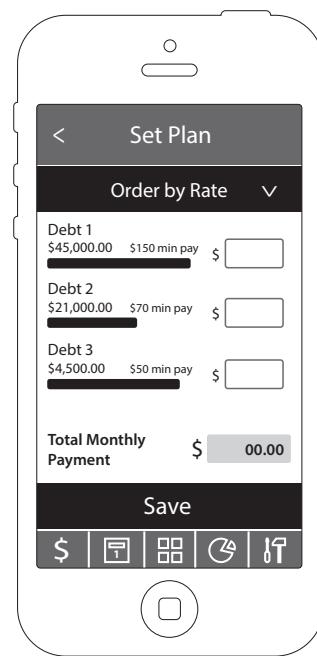
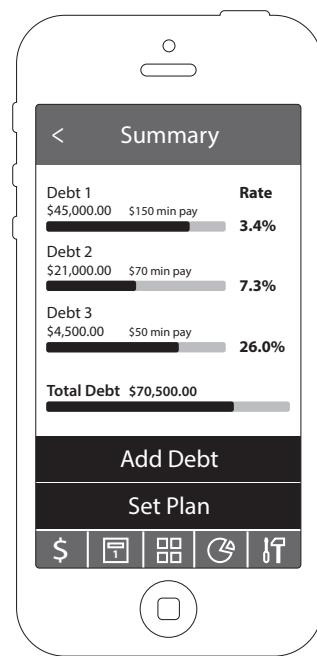
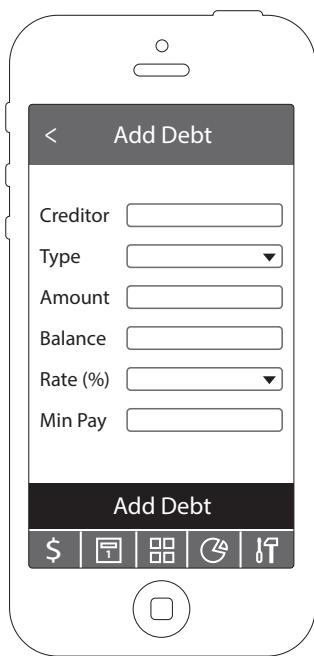
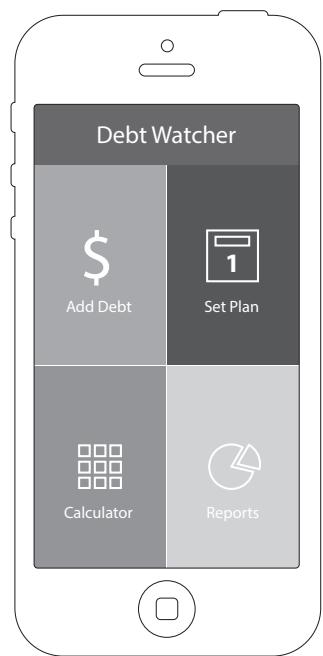
Your app helps you prepare the debt repayment plan.

Set the monthly payments for your debts.

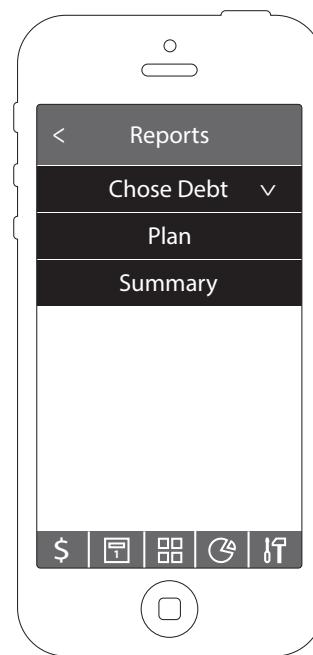
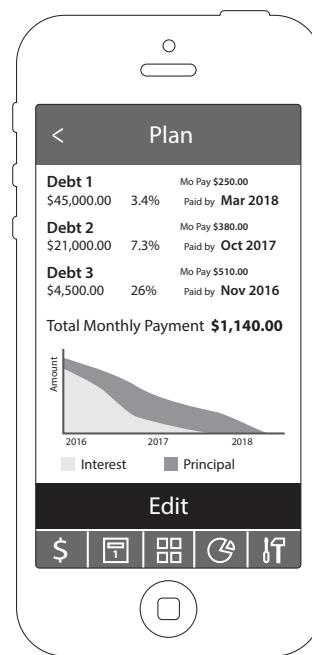
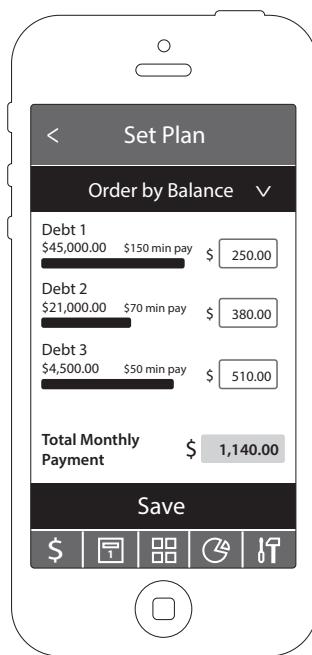
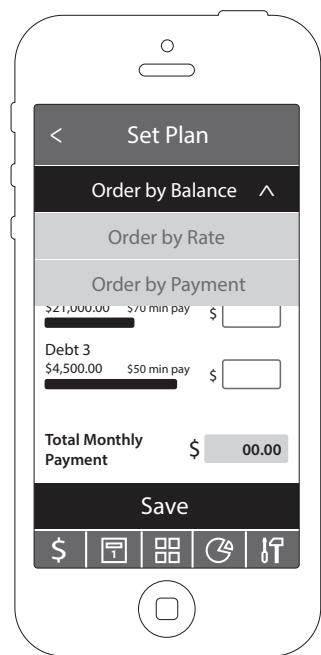
Wire Frame | Wire Framing



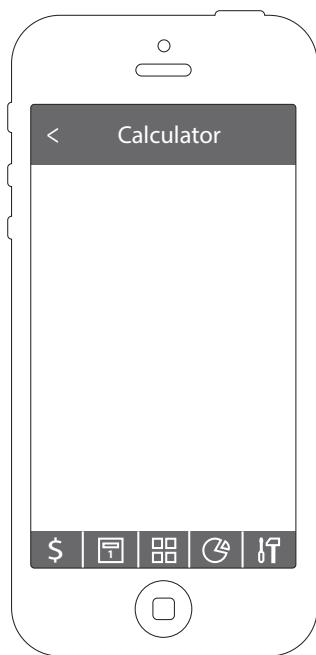
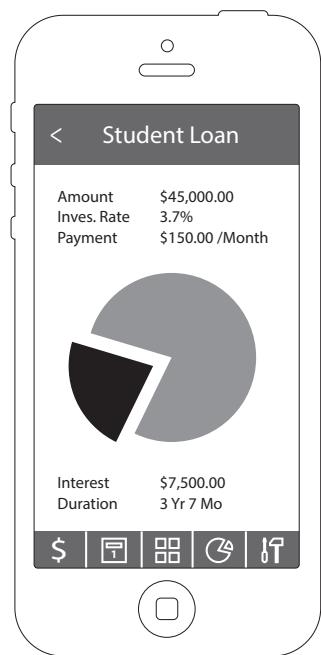
Wire Frame - Details | Wire Framing



Wire Frame - Details | Wire Framing



Wire Frame - Details | Wire Framing



Tests | Wire Framing

Test 1

In the first round I had to adjust some names for something that is more commonly known. For example, I changed the 'Debt Situation' to 'Summary' and 'Statistics' to 'Reports'.

I also changed the 'Payment' value to 'Minimum Payment', as the first value is used to set the plan and was a bit confusing.

The major change was to remove ALERTS section because didn't seem to belong to this app. The impression was that alerts fit to some kind of calendar app. I replaced that with an calculator section that is being designed.

Test 2

After this session I had questions about 'Settings' section which is not designed, but I think that it could be valuable to save/edit some of the values permanently.

I removed the "Duration" (of the loan repayment) from the 'Add Debt' menu because people have a difficulty to calculate that. I thought - what they have this app for? I added 'Minimum Payment' value, as this is important for the calculations. I also included initial amount of the loan.

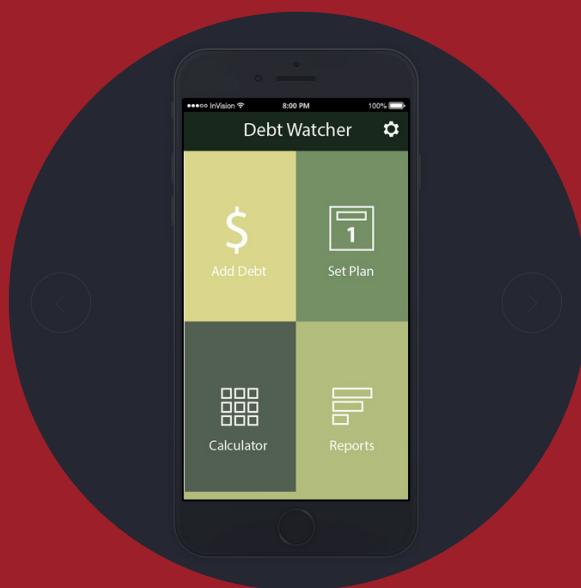
I had suggestions for the calculator section to calculate what if scenarios.

Test 3

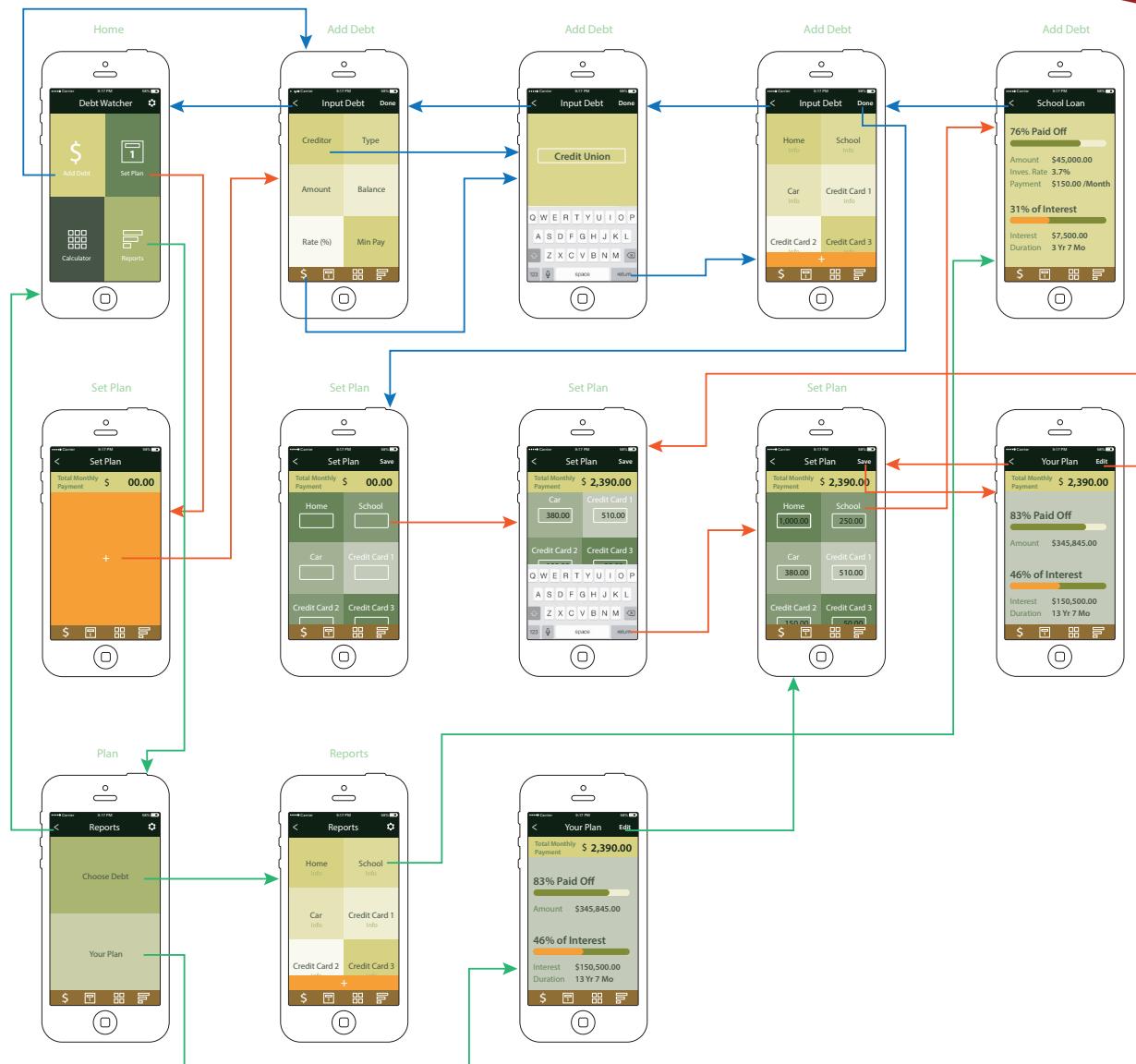
The tests went with no major problems. After some suggestions I prepared changes to the "set Plan" screen. I think that will be more beneficial to include calculations of how much interest will be saved and how fast the loan would be repaid (when the user will input the monthly commitment for that loan) rather than the information that is already known from the previous screens.

Final App

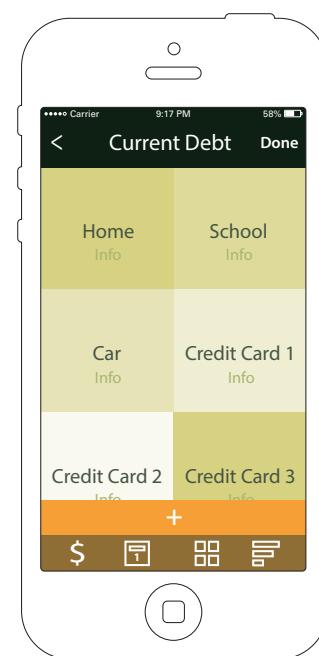
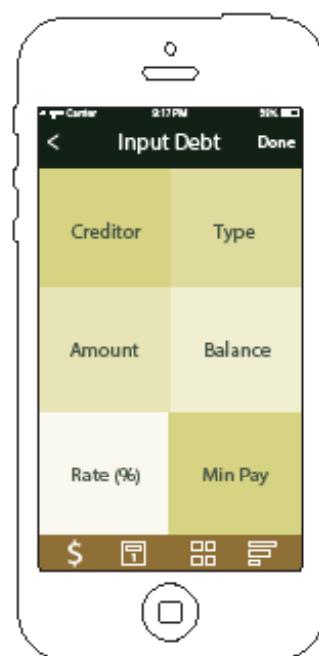
Introduction Research Prototype & Test Wire Framing Final App



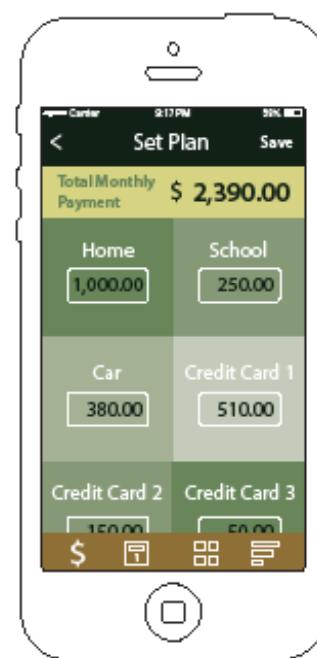
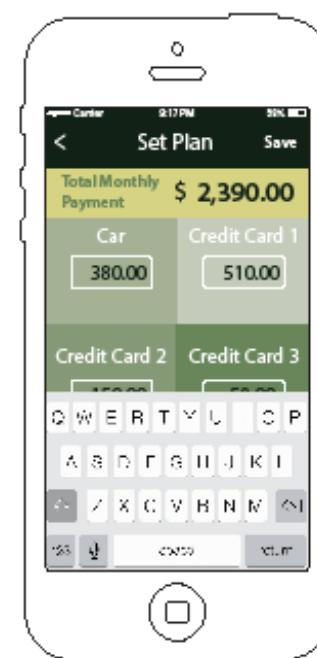
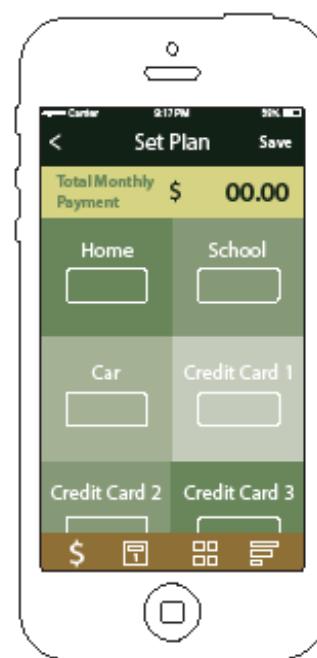
Wire Frame | Final App



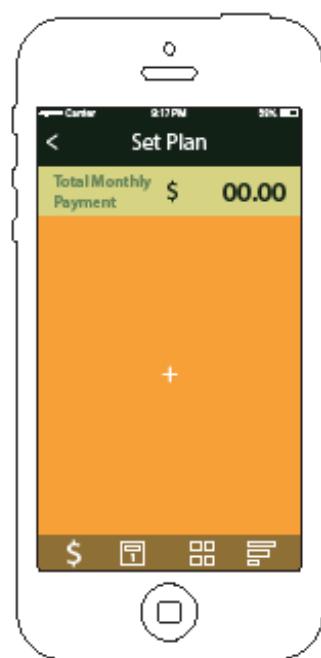
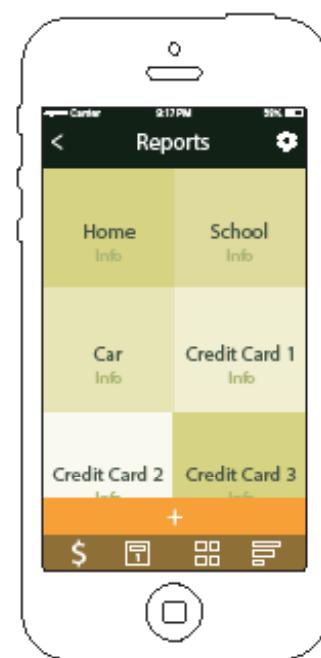
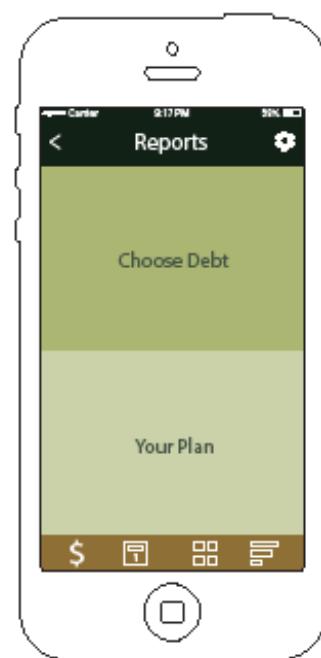
Details | Final App



Details | Final App



Details | Final App



Details | Final App

Test 1

After the first round I simplified actions buttons. I placed them consistently in the upper right corner and named simply 'done', 'save', and 'edit'. The user new by the context what they are for.

I also unified the design to match the home screen. It was a bit of the challenge, but it worked quite well.

Test 2

After some confusions caused by the apparently excessive content, I simplified the amount of information and the way it was displayed. I also unified the graphs, so it feels familiar to the user.

Test 3

I had some questions about calculator and settings functions, but they were not developed at this time.