

Class: CS436 Fall2017

Assignment: HW3

Due time: Sep 21, 2017 11:30 PM

Student: Dong Liang

Professor: Ravi Sethi

TA/Graders: Jacob Combs

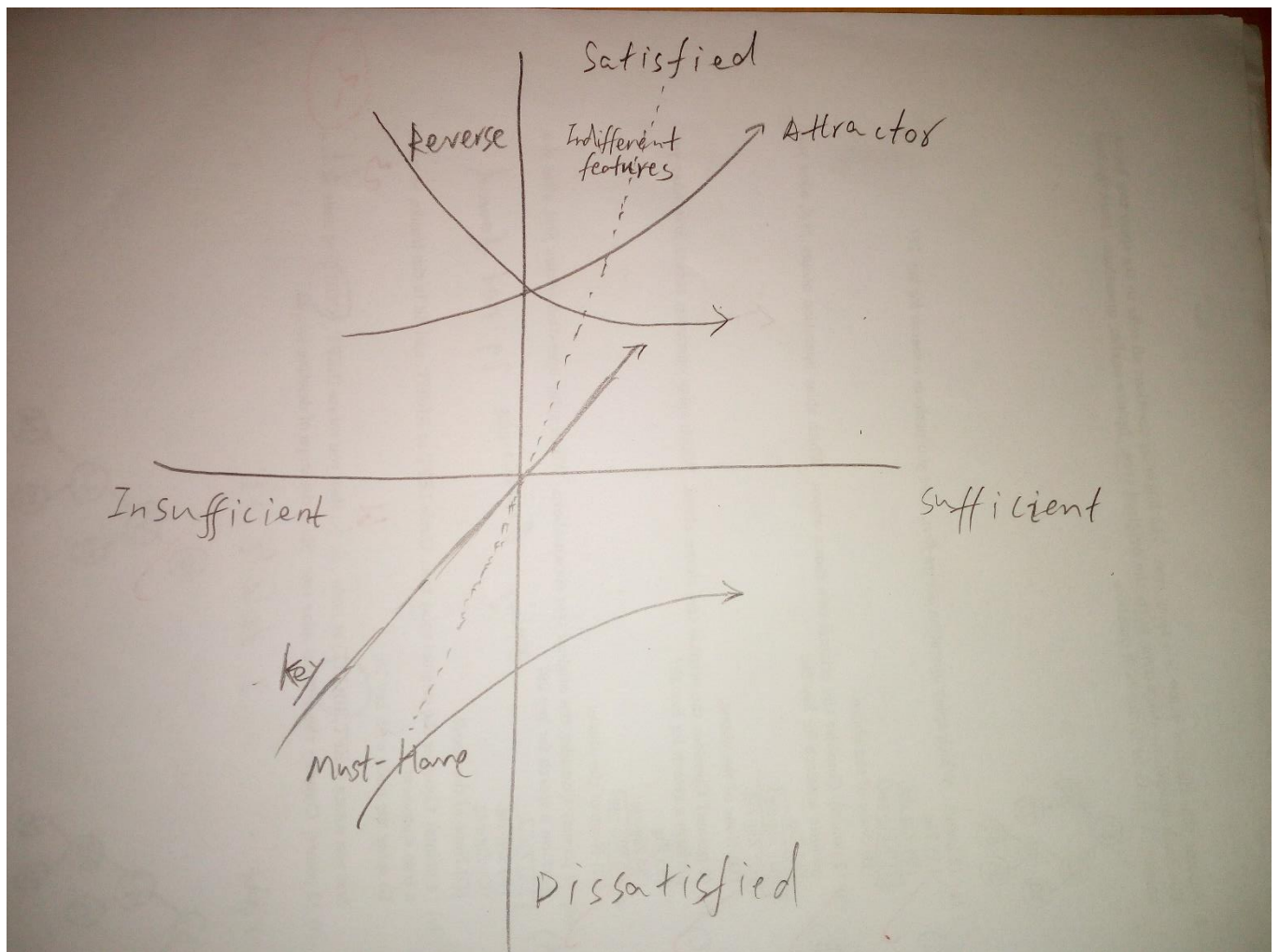
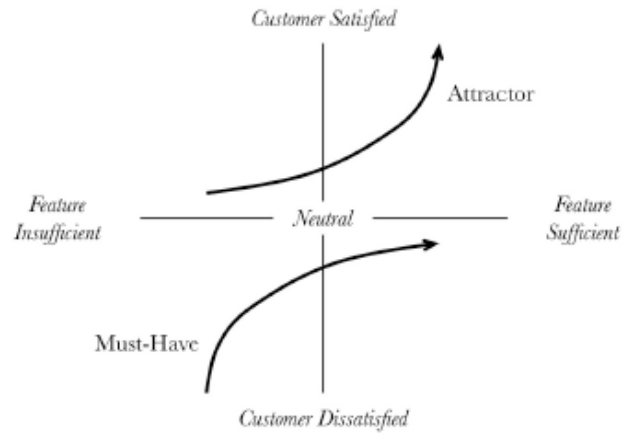
3.1. [10 points] Fill out the following 9-box grid, which is used for the classification of features during Kano Analysis. Give a one-line explanation for why a given class of features belongs in one of the boxes in the in the grid. (Note that the ordering of rows and columns is different from the ordering discussed in class.)

Feature is Built	Neutral			
	Satisfied			
	Dissatisfied			
		Neutral	Satisfied	Dissatisfied
Feature is Not Built				

Neutral	Indifferent	Reverse	Must Have
Satisfied	Attractor		Key
Dissatisfied	Reverse	Reverse	
	Neutral	Satisfied	Dissatisfied

3.2. [10 points] The following conceptual diagram illustrates the increase in customer satisfaction for Attractor and Must-Have features as the sufficiency of a feature increases.

Draw the corresponding curves for Key, Reverse, and Indifferent features.



3.3. [20 points] Write a use case for the software to send a text message between two mobile phones, as described below.

Each phone has its own Home server, determined by the phone's number. The Home server keeps track of the phone's location, billing, and communication history. Assume that the source and destination phones have different Home servers. The destination Home server holds messages until they can be delivered. Also assume that the network does not fail; that is, the phones stay connected to the network.

Your use case must include the following.

- a) A basic flow
- b) Extension points
- c) One specific alternative flow
- d) One bounded alternative flow

In each case, explain how the use case illustrates the relevant concept. For alternative flows, include the full flow, not just the name of the flow.

a) Basic flow:

1. User 1 wrote a text on his/her phone for user 2.
2. User 1 selected the destination user (user 2).
3. User 1 hit the "send" bottom and send the text to his/her number's server.
4. The server identified the user 1's number and the destination number.
5. The server sends only the text message from user 1 to the destination server and send the number of user 1.
6. The destination server holds this message.
7. The destination server checked the text message and send it to the destination number.
8. The user 2 gets the message.

b) Extension points

{compose text}

1. User 1 wrote a text on his/her phone for user 2.
2. User 1 selected the destination user (user 2).
3. User 1 hit the "send" bottom and send the text to his/her number's server.

{send text}

4. The server identified the user 1's number and the destination number.
5. The server sends only the text message from user 1 to the destination server and send the number of user 1.

{check text}

6. The destination server holds this message.
7. The destination server checked the text message
{deliver text}
8. The destination server sends it to the destination number also send a confirm information to the first server.
9. The user 2 gets the message.

c) One specific alternative flow: the destination server blocks the message

1. User 1 wrote a text on his/her phone for user 2.
2. User 1 selected the destination user (user 2).
3. User 1 hit the “send” button and send the text to his/her number’s server.
4. The server identified the user 1’s number and the destination number.
5. The server sends only the text message from user 1 to the destination server and send the number of user 1.
6. The destination server holds this message.
7. The destination server checks the text message and decides to send it back.
8. The first server gets the return text and send a message to user 1 that the text was failed to send.

d) Bounded alternative flow: Note destination server is down”

At any point between {check text} and {deliver text} if the destination server is down, then the first server will not get the confirm information

Server will show user 1 “Something is wrong please send the text again”.

Resume the basic flow at { compose text }