

Happy Customers,
Successful Projects,
& Living Stress Free



Yousaf Shah
Founder / CTO
DataTherapy



What is this talk about?



- The symbiotic relationships between developers and their clients
- Lessons learnt (the hard way!) while running a software business
- What causes of stress for both developers and clients
- Specific things I have observed which trigger success and things which lead to problems
- No-code - relax its the end of the day!

Who am I ?

- Founder / CTO DataTherapy Limited
- Founded 1991 / Team of 14
- From the UK
- Developer, Inventor, Designer, Coder, Boss, Father, Friend, Drummer, Photographer ...



Yousaf Shah

yousaf.shah@datatherapy.com

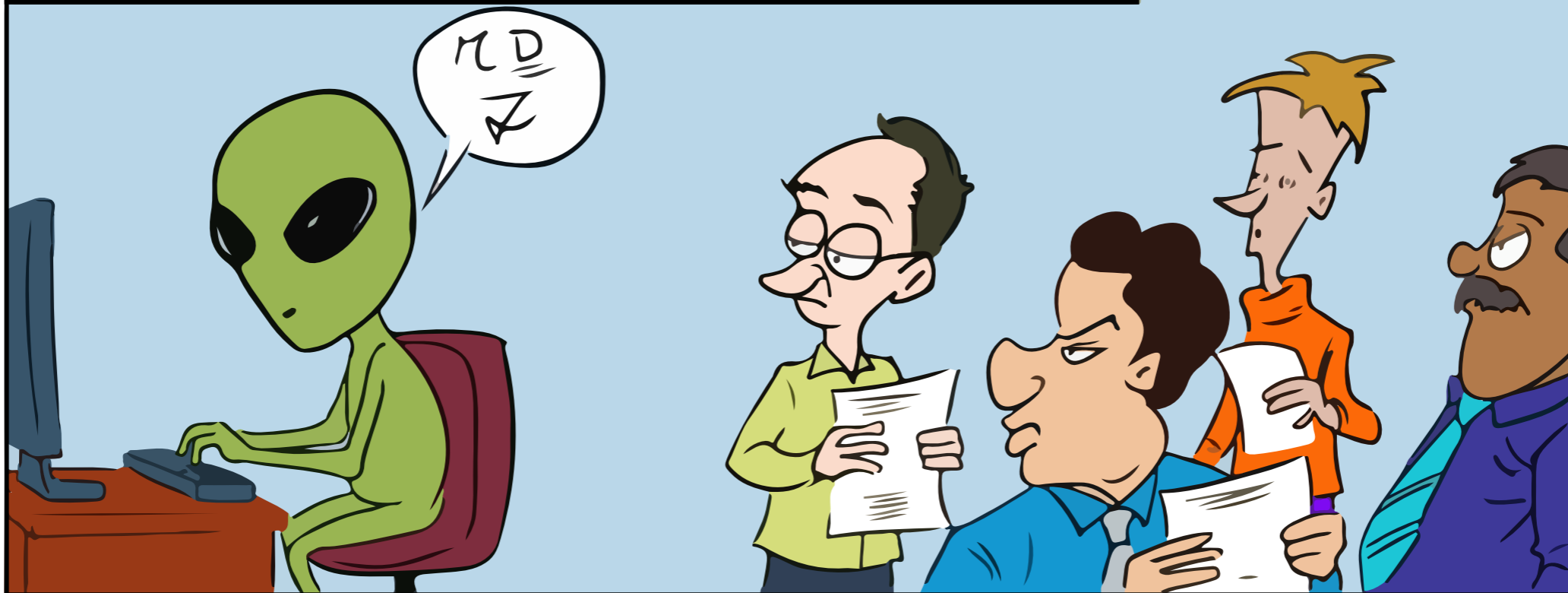


It's just my opinion ... (disclaimer)

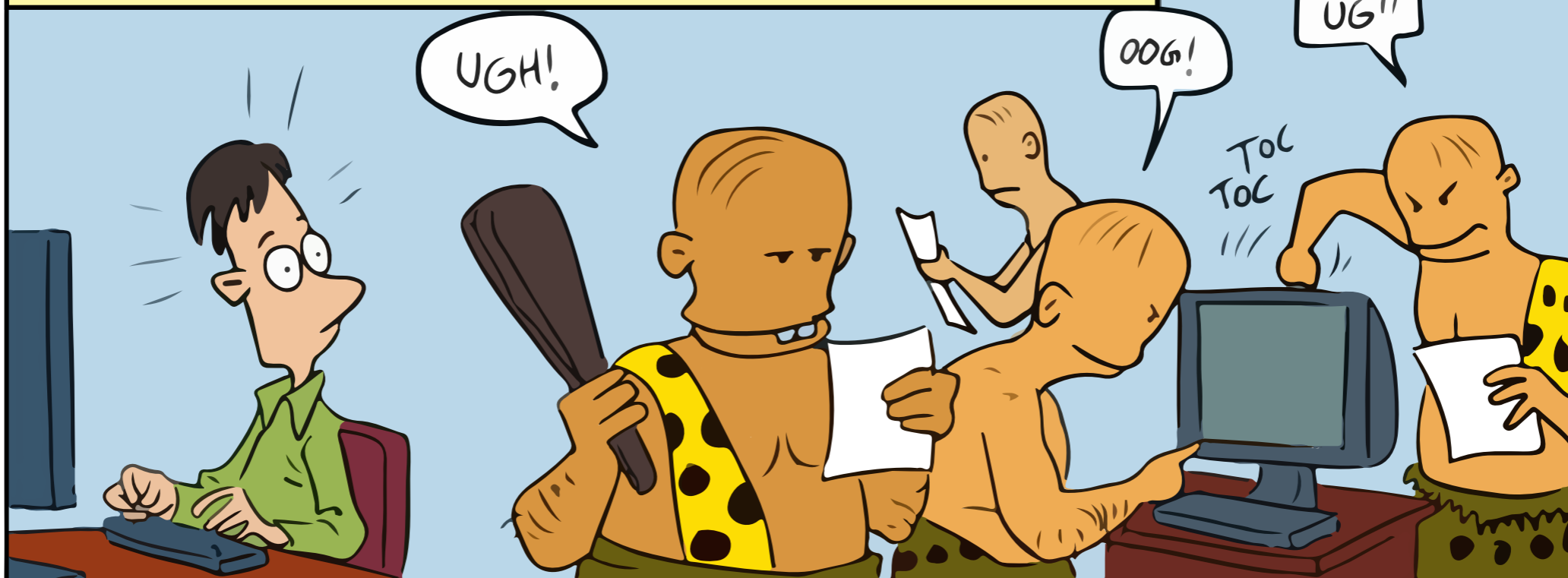
- Everything here is my opinion
- It is not guaranteed to work for you
- You will of course have your own style / experience
- Your clients may not be like mine - but I don't think that matters
- I hope it makes you think a little about your next project, feature, interaction!



HOW USERS SEE PROGRAMMERS ...



HOW PROGRAMMERS SEE USERS ...



Who do we have here?



Develop
for life

Own Company
Freelancer

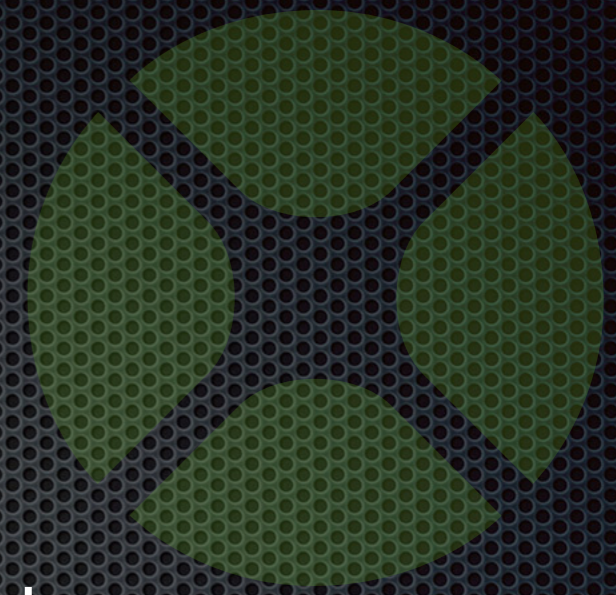
Develop
for wage

Employee
Manager

Develop for
pleasure

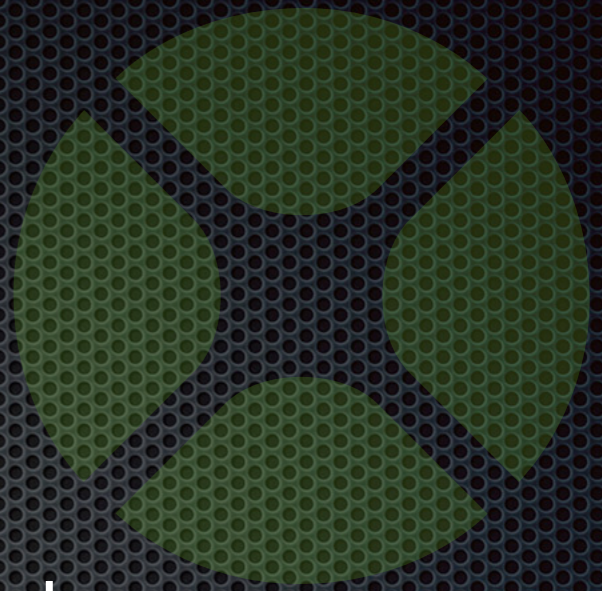
Problem Solving
Personal Projects

We all have 'Clients'



- Whether we run a consulting business, develop freelance or we are paid to sit at a desk and code we all have clients.
- Like it or not, how easy/stressful our lives are depends on how happy our clients are.
- How happy your clients are is, unlikely as we may think, in our control. It is almost entirely down to how we approach our work ...

What Clients want ...



- Understanding what a client is looking for is key. In my experience, they are mostly after the same simple things ...
 - successful project / feature, delivered on time
 - positive experience, minimal stress, pride
 - clear expectations and for them to be met
 - assurance that things are on track
 - to feel that their needs are understood
 - to be able to come back for a new project and make it equally (or more) successful

Happy ...



Things which annoy Clients ...



- What tends to annoy clients are the simple polar opposites, things which mean their wish list is not satisfied:
 - projects which don't work or are delivered late
 - negativity, struggles, fights or embarrassment
 - confusion over expectations, disappointment
 - not knowing how things are going
 - feeling like they have to hand-hold all the way

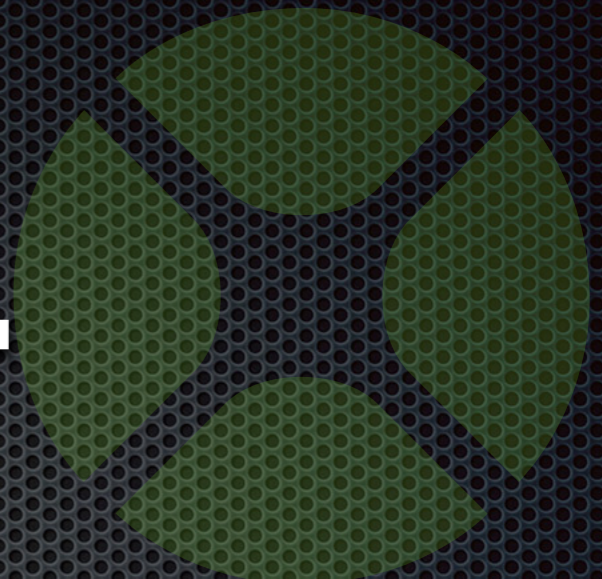
Not amused ...



Stressed Client



It's about empathy ...



- Just remember that the client is usually just trying to get a job done. It's about your empathy towards them.
- They usually have a boss (their client), or they may just have something they are trying to achieve. They are coming to you because they either don't have the skills or they don't have the time to do it themselves.
- Your task is simple - help them get their job done, make them look good, solve their problem.
- Show care and attention for their real task - its not (just) about your software.

What Developers want ...



- Lets not forget what makes us tick, here are some of the things we look for ...
 - successful project / feature, a happy (and interested) client
 - a positive journey, minimal stress, pride
 - a clear specification - build this and you are done
 - to have someone interested and driving for completion
 - to feel appreciated and learn something new whilst developing
 - to be able to finish, get a high-five and start something fresh and exciting!

Happy Developer



Things which Stress Developers ...

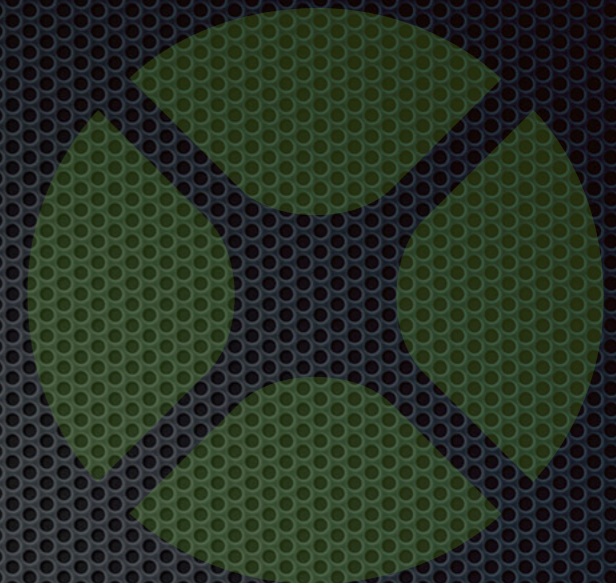


- And some of the things which cause developers stress, as with clients, some of them are the opposites of the previous list:
 - finding out that the client is unhappy, and not knowing what to do next
 - negativity, clients who do not help drive the project
 - having to build things twice / redo work, having the goals changed
 - not being appreciated, feeling like the client does not realise what they are asking for
 - projects which never end, not being able to start the next project because you are still finishing the old one

Stressed Developer



A perfect match!



- When you look at the two sets of lists, it is easy to see that it could be simple for developers and their clients to live happily ever after.
- Clients have a requirement which needs fulfilling, developers get satisfaction out of fulfilling those needs.
- Hopefully you all have some cases where this has worked perfectly, but we know many times it does not work that well.
- Lets look at some simple rules and tools which can help it to work every time.



What can we do ...

Set Expectations



- Agree Specification
 - Get as close as possible to an agreed spec.
 - Talk about how the system will feel
- Agree Pricing
 - By the hour, fixed price, daily rate, hourly rate
 - Most customers have ambitions which outstrip their budget!
 - You set the expectations - think first - speak later ...

Set Expectations



- Agree Timeline
 - Don't need to know necessarily when it will be done, but when you will start, what the dependancies are
 - Find out if there are external demands
- Agree How Often to Meet
 - Its VERY important to keep up constant communication
 - Once per week would be my minimum
 - Gives you something to aim at

and **MOST** importantly ...



- Agree what your client will need to do
 - This is likely critical to any project
 - When they will gather information & give you information
 - When they will test
 - How they will get the software rolled out
- The Agreed Timeline is for BOTH parties

Don't use technology you don't need (or fully understand!)

- We've all done it ... read up on something new (or seen something at a conference) and just thrown it into a project ...
- Interesting because if you have a LOT of general computing experience it can be quite successful
- However a lot of the time it bites you in the a**
- Difficult to predict the results, difficult to tweak, very difficult to support
- Generally safer to use any technique you could call yourself a 'professional' at - well at least 'experienced'!

Be a good listener ...

(Sh** happens!)

- A great client relationship is not about avoiding issues, its about knowing how to solve them when they happen. It's like any relationship
- Let clients air their grievances
 - Do not interrupt them
 - Do not get defensive
 - Do not answer them straight away / dismiss them
 - Hold on to the point where they stop talking ... and pause!
- No matter what you have to say, it will only make them think that you did not listen to them



And when you answer ...



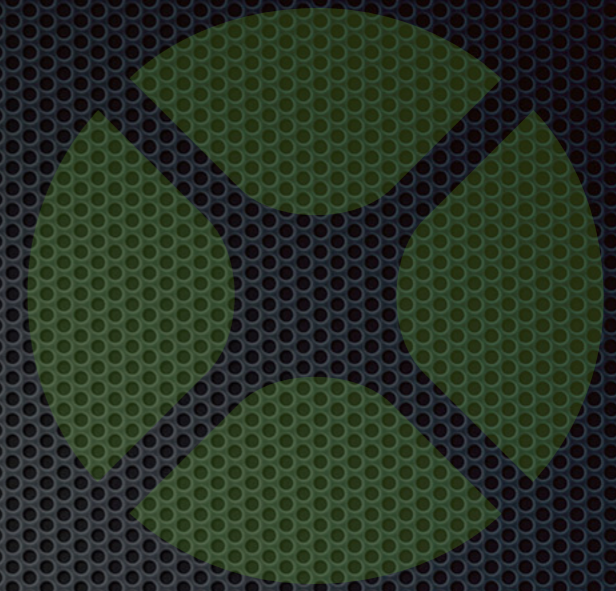
- Best way to respond at the time (empathy)
 - “That does not sound good”
 - “That must have been really annoying”
 - “I will look into it right away”
- Things that are bad to say ...
 - “There is nothing wrong”, “you are wrong to be upset”
 - “It’s not my fault”
 - “It works better that way”
- Don’t get annoyed
- Think about your answer - don’t shy of going back to them the next day

Know when to stop ...

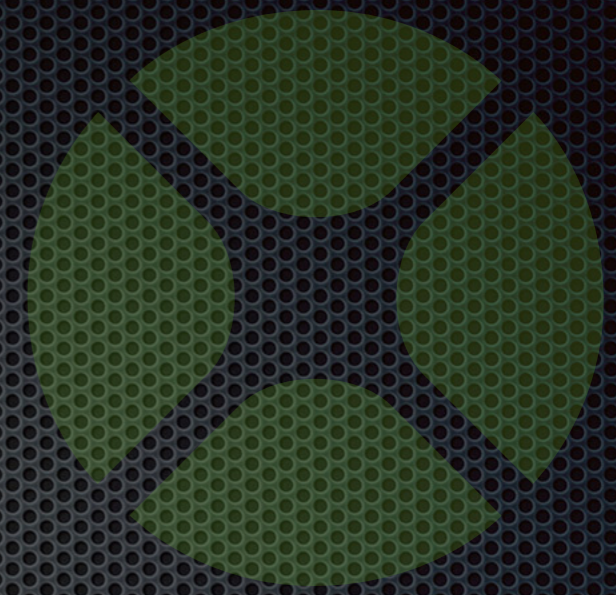


- We all get tempted to add ‘just one more feature’
- Try to stick to the task
- Project Management Software helps
- Over Working Wrecks Relationships
 - Sets false expectations, time, value
 - Makes projects late when they don’t need to be late
 - By all means factor in some time for your ‘pixie dust’

Some general tips

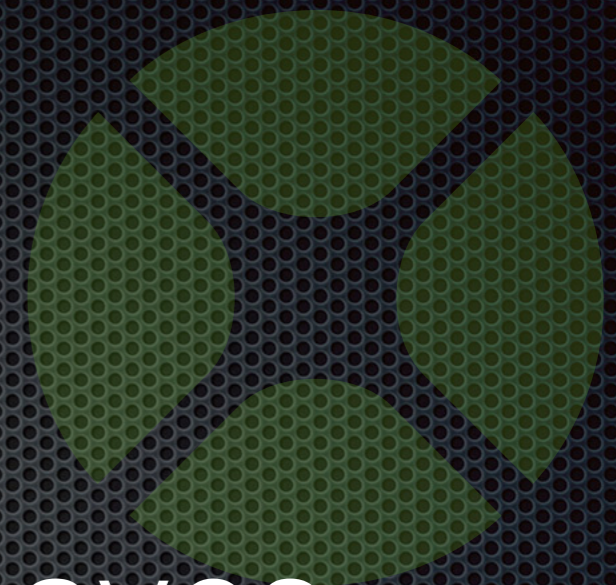


- Always meet a deadline (set realistic deadlines)
 - Cannot stress enough how important this is
 - Everyone has a different attitude to deadlines - don't mess
- Be the person who can (positivity is infectious)
 - Everyone hates that person who is never willing (or moans before he/she does anything)
- Set the right price (value proposition)
 - Everything depends on this!
- And lastly ... communication



“The single biggest problem in communication is the illusion that it has taken place.”

– *George Bernard Shaw*



“Communication always leaves
me incomplete.”

“Communication let me down...”

– *Spandau Ballet*



Would you prefer that I work to an agile manifesto, or would you prefer an acceptance test driven workflow?

I haven't the foggiest idea what you are talking about

When you communicate ...



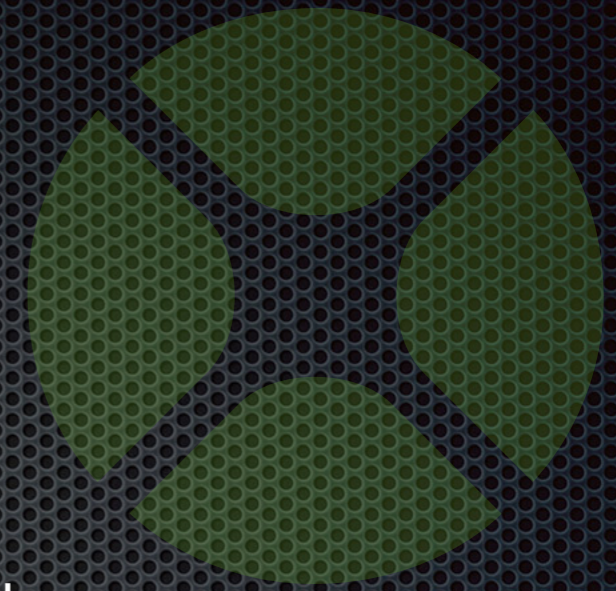
- Try to speak their language
 - Clients love programmers who can explain things in plain language
 - No-one wants to be embarrassed
 - Don't blind them with science
 - Don't talk down to them either
- Try to speak with their goals and point of view in mind
 - Express things in terms of the project, delivery time, their goals
- Don't repeat yourself - try to seek understanding

Specification / Pricing



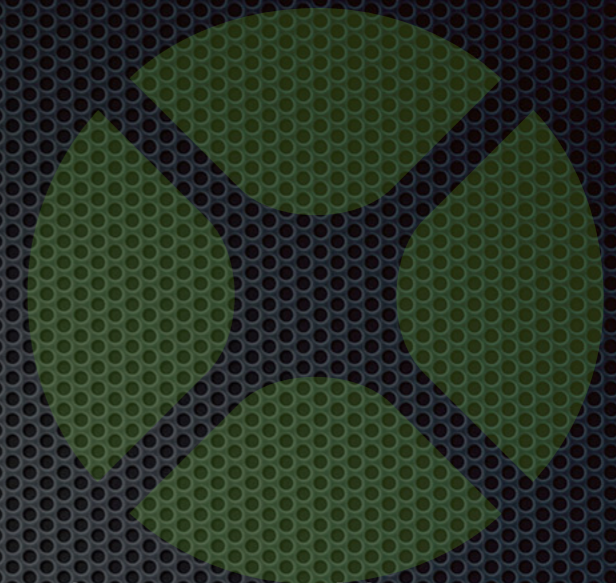
- So a client has asked you to quote for a piece of software ...
- You start to talk about the solution, you come up with a price
- You can be faced with a few problems:
 - client is not sure of the spec
 - client has no concept of difficulty
 - client says he can get the job done cheaper
 - client wants a fixed price
 - client has not developed software before

Lets try it ...



- I have a client who would like a Xojo web project built:
 - For their customer service team
 - To log telephone complaints / enquiries
 - 20 Operators
 - Everyone has a Mac or PC
 - Enquiries will be taken on the phone
- How much would you charge?
- How long would it take?

Traits vs. Features



- It is much simpler to think about a piece of software if you first decide what the traits of the system are
- What are traits?
 - Traits are a way of thinking about how the software, project, experience will feel
 - They are also a handy way of explaining what you are aiming to achieve
 - They can serve to protect all parties

Traits - UI & I/O



No Branding

1. 2. 3. 4. 5.

Fully On-Brand

Raw Interface
(staff training
required)

1. 2. 3. 4. 5.

Intuitive Interface
(little/no training
required)

No Data Validation

1. 2. 3. 4. 5.

Full Data Validation

No Reporting

1. 2. 3. 4. 5.

Full Reporting
Included

Traits - UI & I/O



No Branding

1. 2. 3. 4. 5.

Fully On-Brand

Raw Interface
(staff training
required)

1. 2. 3. 4. 5.

Intuitive Interface
(little/no training
required)

No Data Validation

1. 2. 3. 4. 5.

Full Data Validation

No Reporting

1. 2. 3. 4. 5.

Full Reporting
Included

Traits - Project / Rollout



Client Liaison
Collects Information

1. 2. 3. 4. 5.

Developer Collects
the information

No Documentation

1. 2. 3. 4. 5.

Full Documentation

No Training

1. 2. 3. 4. 5.

Training
for all staff

No warranty/
all fixes paid

1. 2. 3. 4. 5.

Long term warranty/
bug fixes

Traits - Project / Rollout



Client Liaison
Collects Information

1. 2. 3. 4. 5.

Developer Collects
the information

No Documentation

1. 2. 3. 4. 5.

Full Documentation

No Training

1. 2. 3. 4. 5.

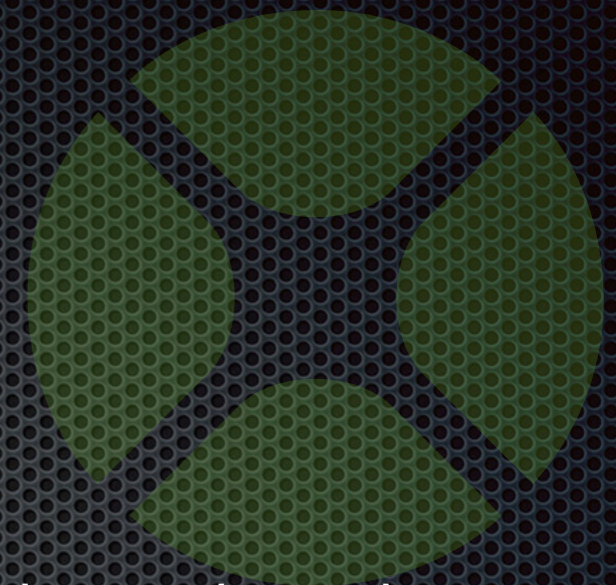
Training
for all staff

No warranty/
all fixes paid

1. 2. 3. 4. 5.

Long term warranty/
bug fixes

Traits can really help

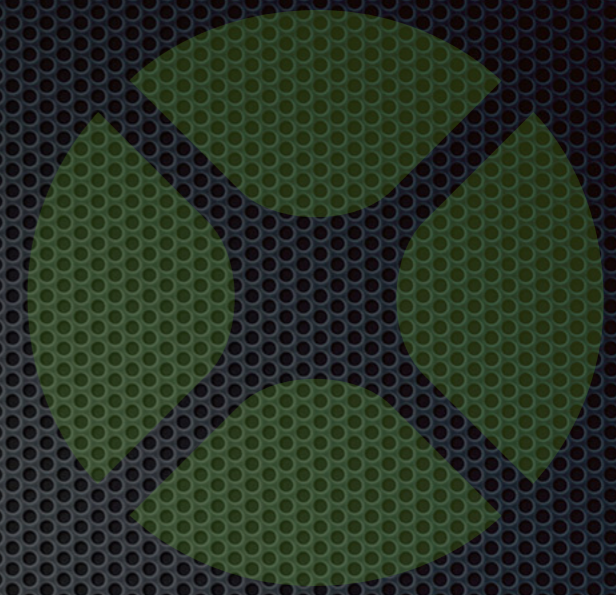


- If you supplied a sheet with your quotation explaining what the traits of the system are that you are quoting:
 - Gives an immediate benchmark
 - Stops them coming back saying 'I did not expect ...'
 - You know when to stop, how far to go
 - You can get help and effectively tell them what is expected
 - You get a chance to bargain over price with real information
 - Clients can also ask you to dial up the trait - and pay for it

Finally ...

Rate your client

- Has the client ever developed software before?
- Is your project dependant on the client doing some work?
- Is the client willing to test the software thoroughly?
- Is this project really important to them?
- Is there enough business drive behind the project?
 - Is the CEO, CIO, CFO involved?
- Is there someone who can be your main project contact
 - Do you think they can do the job?



Thanks for listening!

Yousaf Shah

yousaf.shah@datatherapy.com

Give us feedback on this session in the XDC app!

