MINISTRY OF EDUCATION AND SCIENCE OF UKRAINE
HIGHER EDUCATION ESTABLISHMENT
"UKRAINIAN CATHOLIC UNIVERSITY"

Department of Social Sciences
Chair of Management and Organizational Development

Master's thesis
Entitled: "Establishing result-oriented presales process at Sombra"

Performed by: 6th year student, group STM17/M
speciality 073 "Management"
Nakonechnyy Yuriy

Supervisor: Yevhenii Matvieiev
Reviewer: Vasyl Matiashovskyi

Lviv 2019

Table of contents

Introduction

In July 2018 top management of Sombra company faced problem with monthly revenue that stopped growing. This happened because some projects ended and flow of new projects generated by sales wasn't enough to cover growing demand. This contradicted with company's yearly strategy goals because according to them monthly revenue was supposed to grow until the end of the year 2018. The problem persisted from July 2018 till December while Sombra top management team was trying to find out the reason for it. After digging into the details two causes were discovered:

1. Insufficient number of leads entering sales pipeline
2. Some leads not converting into closed deals at different stages of sales pipeline

Chief Sales Officer - Sergii Miakshynov - was appointed to solve problem with number of leads by aggressively diversifying sales channels. Being a CTO at Sombra, I decided to study ways on how to increase efficiency of presales process in order to maximise revenue coming from incoming leads. This was relevant both in short and long term perspectives especially because presales process wasn't improved much during last couple of years while outsourcing industry become saturated with competitors.

It was decided to bring more attention to presales process starting from December 2018 and thoroughly examine most presales cases, especially lost ones.

Business context of Sombra

Sombra was founded in the end of 2013 by CEO - Viktor Chekh. Later in 2014 two other co-founders joined - Chief Sales Officer Sergii Miakshynov and me as Chief Technology Officer. Together we formed top management team that made decisions up until now. General vision of Sombra was formed as "Growing reliable software company. The best people to work with." meaning that company must always grow and doesn't stop on some certain size. Also, from the start Sombra heavily relies on its 5 values that were carefully carved out and discussed many times. These are:

1. Customers: they are at the core of our business. Our goal is to help develop their business to its full potential
2. Proactivity: we welcome people who take initiative and responsibility for their lives and actions
3. Open and honest relationships: we build open and honest relationships with colleagues and customers. We do not let artificial barriers in communication get in the way and we always speak our mind if there is something wrong
4. Changes: our company is growing rapidly, which is why changes are inevitable. We must easily adapt to changes
5. Self-development: each of our employees has to grow both professionally and personally - this is a precondition of our successful cooperation

All employees are selected by sharing these values and if person's internal beliefs heavily contradict any of them - we doesn't hire this person even if from technical standpoint this person fits certain project and customer well.

From the start in 2013 / 2014 Sombra heavily relied on Upwork as sales channel providing constant flow of hot leads. Two major directions were formed at that time:

1. Building turnkey solutions
2. Providing dedicated teams

Engineering pool led by CTO was divided into 3 major directions from the start of 2015:
1. Java engineers (building backends)
2. JavaScript engineers (building web and hybrid mobile frontends)
3. QA engineers ensuring quality of end product

About a year after start we decided to build Project management expertise, which by 2018 grown to 4 Project managers who are responsible for delivery of projects in terms of scope / time / budget.

Growth of company was quite intensive during 2013 - 2018 starting from 100% to 40% year-over-year growth. This was due to the fact that Upwork was providing big number of hot leads and company lived in constant shortage of engineers rather than new projects. During this period of organic growth company formed organizational structure with clear segregation of responsibilities.

1. Chief Executive Officer
    a. Delivery Director
        i. Project managers
            1. Project teams
        ii. UI / UX designers
    b. HR Director
        i. HR managers
        ii. HR marketing manager

        iii.   Event manager

        iv.   Recruiting managers

    c. Head of Finance & Legal

        i.   Accountants

    d. Office management

2. Chief Sales Officer

    a. Head of Marketing

    b. Engagement managers

    c. Sales managers

    d. Account managers

3. Chief Technology Officer

    a. *(indirect) technical leads of major directions (Java, JS, QA)*

    b. System administration

This structure was setup historically to be able to scale to growing number of projects and people inside company. One could also refer to growth statistics of Sombra during last 5 years:

NUMBER OF PEOPLE

2013	2014	2015	2016	2017	2018	2019
3	10	28	60	90	120	170
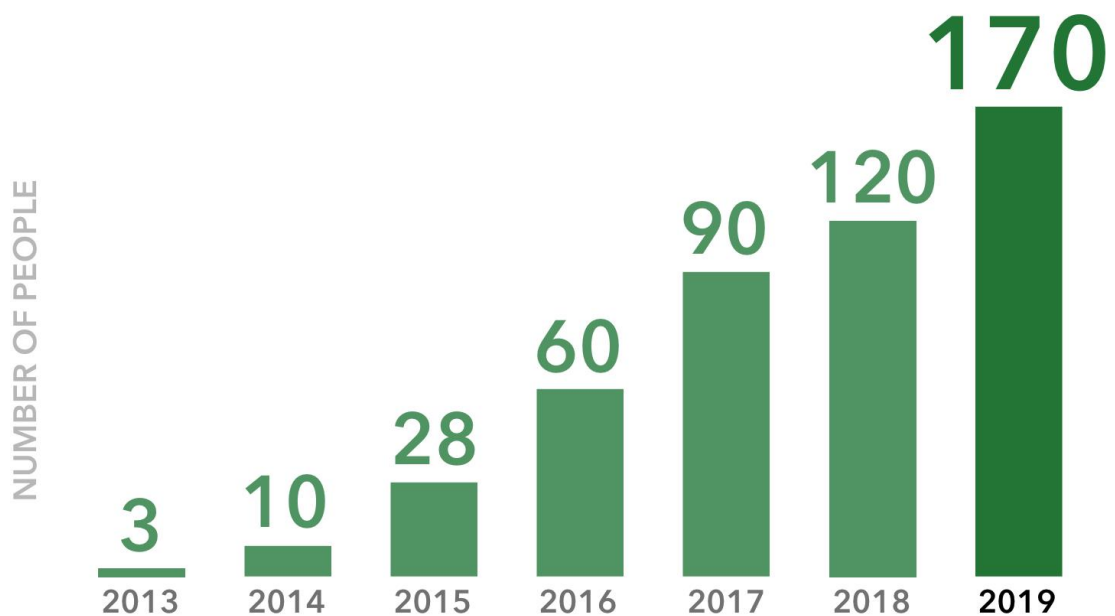
Size of engineering department being one of the company's valuable assets (by means of revenue generation) grew proportionally to size of company and by the end of 2018 it consisted of about 80 engineers:

1. 40 backend engineers (Java / Kotlin + wide data storage / cloud experience)
2. 30 frontend engineers (JavaScript / TypeScript and related languages for building web and hybrid mobile frontends)
3. 10 QA engineers (manual and automation QA engineers)

By the end of 2018 Sombra had around 20 customers ranging from smaller ones working with 1 engineer from our side to large ones with 20 engineers worked in dedicated teams for one customer. Most of those customers came from Upwork sales channel however some of them also approached us via website directly.

Product management perspective

To be able to better understand our service it makes sense to analyze our services using Product Management tools to understand what problems of which client segments we solve, what are doing our competitors and what will we do in a year. This section is heavily inspired by Product Management Course of Scott Sehlhorst.

Theodore Levitt's Total Product Model

"The customer rarely buys what the company thinks it is selling" - Peter F. Drucker

Before jumping into client segments and competitor's comparison we first must understand problems that we are solving for our clients. Theodore Levitt's provides following framework for defining product differentiation:
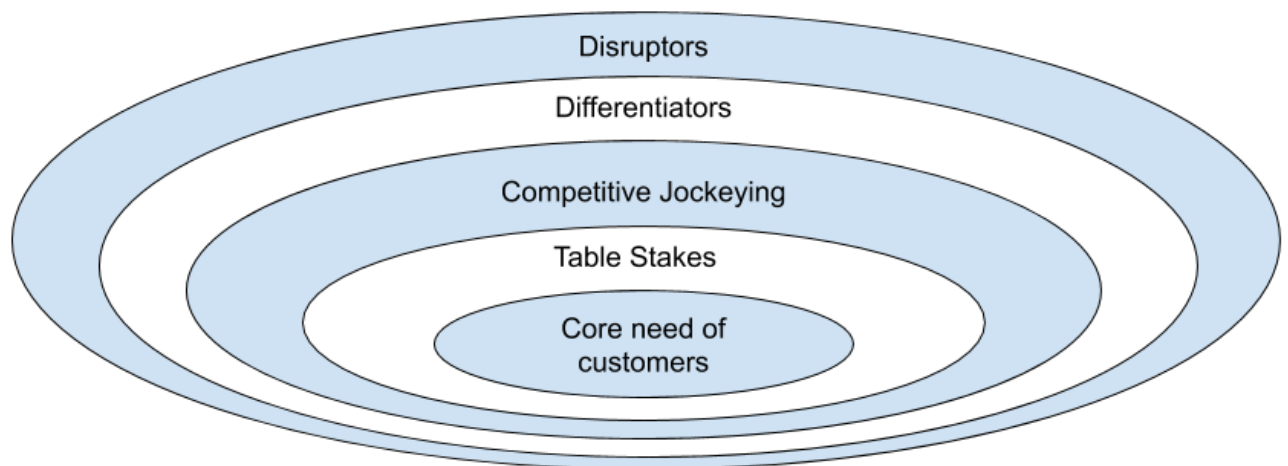
1. Core need of clients

   In software outsourcing market it's the need to have a software engineering team that is able to deliver desired software

2. **Table Stakes** (or Generic Product in Theodore Levitt's terms)

   a. 40 hrs / week

        b. sufficient level of expertise

        c. Intermediate level of English

3. **Competitive Jockeying** (or Expected Product in Theodore Levitt's terms)

        a. Hourly rates

        b. Speed of staffing

        c. English level

        d. Seniority level

        e. Timezone

4. **Differentiators** (or Augmented Product in Theodore Levitt's terms)

        a. Proactivity ("not to be like robots", suggest better ideas, advise)

        b. Team leading ability: could be managed as single unit

        c. Team integration ability: ability to mix software development teams by organizing two-way business trips

        d. Ability to take bigger responsibility: commit to deadlines, scopes and/or budgets

        e. Multiple stack expertise

        f. Niche technology expertise: blockchain, ML, AR / VR

        g. Niche domain expertise: healthcare, fintech, other

        h. High level of trust & reputation

5. **Disruptors** (or Potential Product in Theodore Levitt's terms)

        a. Ability to take even more responsibility - provide digital solutions to business problems by performing business analysis and owning the whole scope

First 2 types of product (table stakes, competitive jockeying) are also called generic and expected products - those are offerings **not being talked about**, because they doesn't differentiate much from other products.

Second 2 types of product (differentiators, disruptors) are also called augmented and potential product - those that exceed customer expectations and making the most of the product. These are the products that excite everyone and ones that are **most talked about**.

This was the framework for **problem framing** i.e. how we as a provider decide which client problems should we address.

Noriaki Kano analysis

Knowing customer problems and addressing them is not enough - next important step is **problem characterization** i.e. trying to understand how client thinks about problems he's trying to solve. In other words, how our clients judge our software development services or services in general.

For this Noriaki Kano analysis is applied - a theory of product development and customer satisfaction.
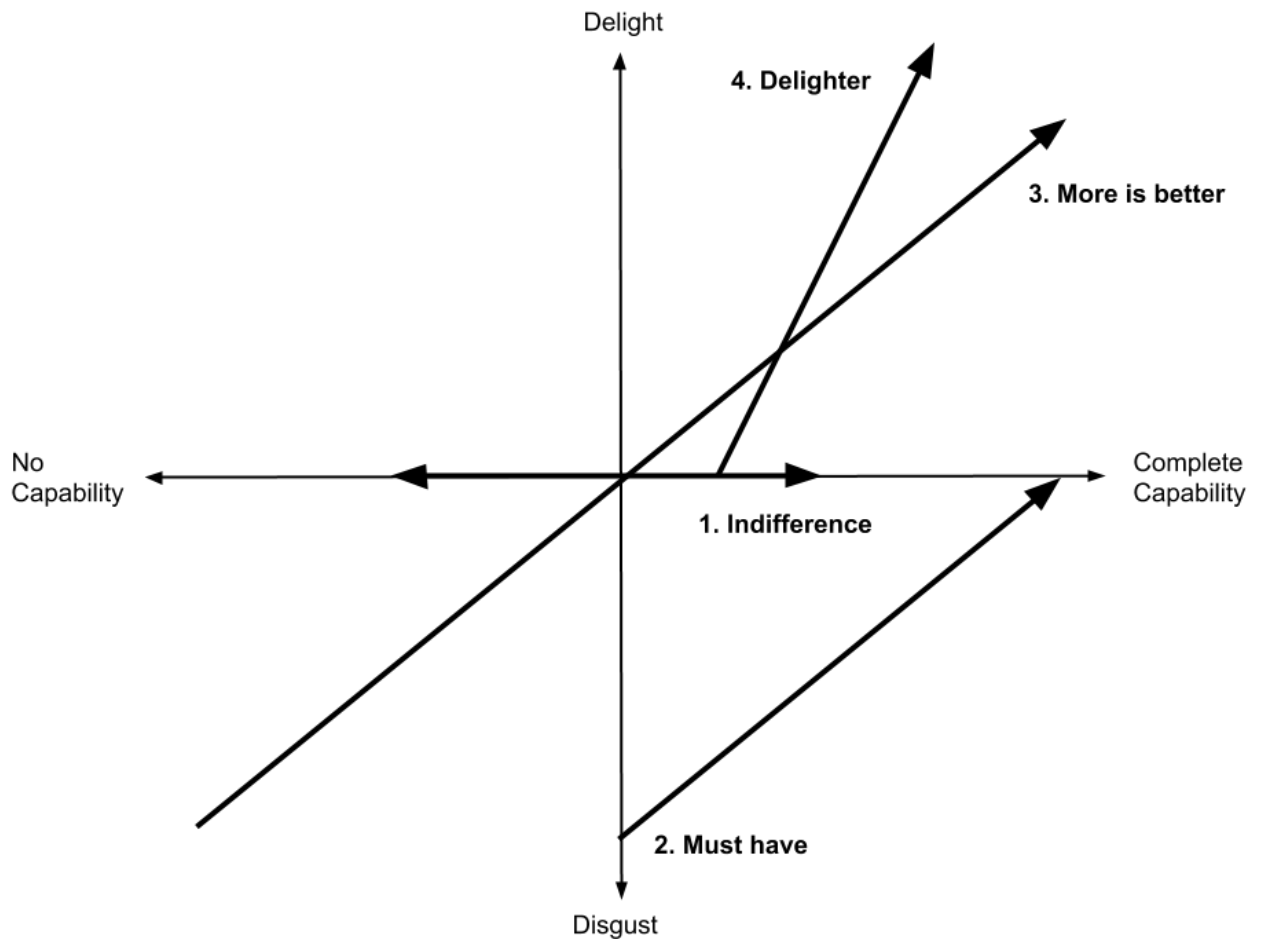
Kano classifies customer's feeling about the problem into 4 groups:

1. Indifference or "I don't care if you solve this problem for me"
2. **Must-have** or "if you don't solve this problem, I'm not interested"
3. **The more the better** or "the better your solution the more I like it"
4. **Delighter** or "Unexpected and extremely good"

Let's analyse perceiving of problems by customers using Noriaki Kano analysis:

1. **Must-have**
   a. 40 / hrs week
   b. English level - intermediate
   c. Sufficient expertise

2. **The more the better**
   a. Hourly rate
   b. English level
   c. Speed of staffing
   d. Expertise level
   e. High level of trust & reputation

3. **Delighter**
   a. Proactivity
   b. Team integration ability: ability to mix software development teams by organizing two-way business trips
   c. Ability to take bigger responsibility: commit to deadlines, scopes and/or budgets
   d. Multiple stack expertise
   e. Niche technology expertise: blockchain, ML, AR / VR
   f. Niche domain expertise: healthcare, fintech, other

This analysis shows us how clients on the market perceives each problem however this doesn't tell us which problems we should address in the first year, second year and so on. In order to do this a tool named "Competitive matrix" developed by one of our lecturers Scott Sehlhorst will be used.

Competitive Matrix analysis

In order to determine which problems we must solve to compete effectively and how to make our service the "best" choice, we need to go through following stages:

1. Identify client groups
2. Define relative importance of customers
3. Identify important problems
4. Identify importance of solutions to customers
5. Identify competitors and alternatives
6. Predict future levels of capability of competitors
7. Identify target capability levels (long term)
8. Identify target capability levels (near term)

Identifying client groups

"A product designed for every customer is designed for none of them"

Our company operates in B2B sector serving mainly customers from US, so I'll start identifying client group by US market segments relevant to our business.

Out of our experience there are following client groups that need software development:

1. **Early stage startups**

   These startups usually consist of one or several founders (with someone technical or everyone non-technical) looking for company to develop MVP andmaintain and extend it afterwards while it'll get traction. Their budgets usually range from 20k to 100k and they need MVP to be delivered within this budget.

2. **Mature stage startups** (rounds A/B or later)

These usually have several founders and some technical team and look for extension of their software development team. They usually have budget for the team of 5 - 10k / month for 3-9 months before their next investment round.

3. **Small consulting companies**

   These usually are companies of size 3 - 20 people that provide software consulting services to businesses and enterprises in their area and need software development team to help them build software products that they deliver to their clients as part of digital transformation consultancy or the like. These companies usually have budgets dedicated to certain consulting project for their customer and this could vary greatly in duration (from 2 weeks to 6 months or longer)

4. **Established product companies**

   These are established businesses whose main product / service is software-based, which have established and working business model relying on paying customers. They need to extend their software development teams usually to give away non-critical pieces of software (PoCs / MVPs) to be built for them, while their teams focus on main products. They usually have dedicated budgets for these certain software projects which could vary greatly in timeline - from 2 weeks to 1 year or longer.

5. **Large enterprises**

   These are departments in large enterprises (banks, telecommunication companies) who look for software developer teams to extend their capabilities. These have established selection procedures for working with vendors and have either budgets for separate projects or for long-term time & material work

Defining relative importance of customers

Given our strategy goals for 2019 which amongst other underline the following:

1. Departure from Upwork as main sales channel
2. Achieve 30% growth of our engineering pool

During quarterly strategic meeting with Sombra owners in the beginning of March 2019 we identified following customer attributes to pay attention to:

1. **Long-term commitment** (1+ years) from customer to have dedicated team
2. Ability to pay **competitive monthly rates** which allow us to attract better talent and thus fulfil customer's needs in the best way
3. **Ease of entry**: easy of finding and starting working with such companies
4. **Account management complexity** of such companies (if companies are big and we don't know key decision makers and budget holders, then this means we doesn't know plans for this and next years and may miss opportunities / observe risks too late)

Let's assign numbers to attributes on the scale of 1-10 for each customer.

1. **Established product companies**

   These companies are more stable in terms of allocated budgets and 12+ month plans, so we estimate them as 10/10 for long-term commitment attribute. Also these companies are quite innovative in terms of technology and agile processes, making integration of our teams into their processes more efficient and working as a win-win situation. Because they are profitable US companies their budget is not very limited usually, so we put 9/10 for competitive monthly rates attribute. From the ease of entry perspective, they aren't as easy to approach as early startup companies and usually face-to-face meeting is required to start working with them, so we put 6/10 for ease of entry attribute. Because they aren't very big companies

having up to 40 people, it's relatively easy to reach out to CTO or VP of Engineering positions, that's why we put 8/10 for account mgmt simplicity.

2. **Small consulting companies**

These companies usually have stable flow of work from existing or new clients and have a continuous need for development. The problem is that software development need could be unstable throughout a year, however it's rarely that it unexpectedly stops in 1 month, because business model is quite predictable and stable that's why these companies are estimated as 5/10 in terms of long-term commitment. In terms of monthly rates these companies usually have limited budgets and calculate them more on task-by-task basis so while hourly rate is bigger, overall monthly rate averaged to yearly period is lower, that's why these companies are estimated as 5/10 with regard to competitive monthly rates attribute. These companies are easier to approach since they aren't big hence the mark is 7/10. Account mgmt with such companies is very simple because of their size, that's why 9/10 is the estimated mark.

3. **Mature stage startups**

These are startups that past MVP stage and acquired investments for next 12 months to continue customizing their product / service and look for product-market fit. Some of them have very promising ideas backed up by solid investments, however they rarely could commit even to 6 months duration and if anything goes wrong - usually contracts with vendors are the first to be suspended, hence mark is 5/10. Most such companies have quite limited budgets and thus they try to save on offshore software development hence mark is 7/10. When these companies have a need - they are very easy to approach, hence ease of entry mark is 8/10. Such companies have usually up to 20 people so account management is simple as well.

4. **Large enterprises**

   These companies are the most stable in terms of long-term commitment and they usual minimum contract is 1 year long, that's why we put 10/10 for this attribute. Regarding monthly rates these companies usually involve their procurement offices and have several other approved vendors. So when it comes to monthly rates they look at a bare numbers, compare with other vendors on paper and drive monthly rates down to the point where it becomes challeging to provide best talent for them, hence mark for this attribute is 5/10. Because of their size, such companies have large organizational structures where it's not often clear who's decision maker and who's not and very often even if they have needs they also have list of big approved vendors which are already working on closing that needs. That's why entering such companies is hard at this moment so the mark is 1/10. Same goes for account mgmt which is very hard due to very distributed structure and limited visibility of vision towards your co-operation - mark is 3/10.

5. **Early stage startups**

   At this phase startups are usually very unpredictable and 95% of them doesn't go past MVP stage, that's why we put 2/10 in terms of long-term commitment. They are also very tight on budget, so competitive monthly rates attribute is set to 3/10 out of our experience. It's quite easy to start speaking with these startups hence ease of entry is estimated as 10/10 as well as account mgmt simplicity is set to 10/10 because they usually consist of several founders.

So, performed analysis leads to following matrix:

Attribute of customer	Established product companies	Small consulting companies	Mature stage startups	Large enterprises	Early stage startups
Long-term commitment	10	5	5	10	2
Competitive monthly rates	9	5	7	5	3
Ease of entry	6	7	8	1	10
Account mgmt simplicity	8	9	8	3	10
	33	**26**	**28**	19	25

Given this analysis, strategically we selected following 3 customers and each of them is currently has almost same priority for our company, except for the first one:

1. Established product companies - 4/10
2. Small consulting companies - 3/10
3. Mature stage startups - 3/10

So further analysis will focus only on these 3 categories.

Identifying important problems

1. **Established product companies**

   Main problems for these companies are the following:
   - **Proactivity**. These companies require high proactivity from developers in order to have mutually beneficial co-operation - this compensates for big timezone shift, cultural differences and distributed team management issues.

   Importance of this problem to this segment is assigned as 10/10

- **English level** (intermediate high or higher). These companies used to work with inhouse developers and that's why they expect English level to be high and this is a must rather than nice-to-have or delighter feature.

  Importance of this problem to this segment is assigned as 8/10

- **Expertise.** Out of our experience, our developers with relevant diploma (e.g. applied mathematics) and 4+ years of relevant experience are providing great value when managed by their Tech Lead developers. Such companies usually face tough competition for talent with giants like Google, Facebook and others and that's why it's hard for them to attract and keep bigger talented teams, so they focus on keeping strong core development teams and struggle with extending development teams.

  Importance of this problem for this segment is 7/10

- **Staffing speed.** Ability to provide team of 2-5 developers in 1 month and grow it to 5-10 developers in 2 months. Such companies could wait for 1-2 months to staff a team, however not more.

  Importance of this problem to this segment is assigned as 6/10

- **Stability**. As onboarding of new team members takes time, they expect developers to stay with them for minimum of 1 year or longer.

  Importance of this problem to this segment is 9/10

- **Trust**. For most product companies such partnership is usually very important one, so they are very discrete in how they choose their partners. They'll divert 99% of cold email / Linkedin proposals even if their value proposition clearly matches their need, due to absense of trust. Such companies will look for references and recommendations from their network rather than online promises

Importance of this problem to this segment is 10/10

2. **Small consulting companies**

   - **Proactivity** (same as with Established product companies)

     Importance of this problem to this segment is 8/10

   - **English level.** Intermediate high or higher (same as with Established product companies)

     Importance of this problem to this segment is 6/10

   - **High-flexibility** in terms of scaling team up or down. Such companies usually work with bigger clients which tend to delegate really urgent work to them so this is projected on our staffing respectively. Team may grow to 6 people in 2 weeks and then shrink to 2 people again in 2 months

     Importance: 8/10

   - **Staffing speed.** Here they may be a need to scale teams faster - ability to provide team of 4 developers in 2 weeks' time

     Importance: 9/10

   - **Trust** (same as with established product companies)

     Importance: 10/10

3. **Mature stage startups**

   At this stage startups mostly need only Software Development capability, doesn't need Business Analysis expertise and more rarely need UX design or Project Management expertise, because they prefer to keep most of them inhouse. So problems are very much like with small consulting companies

   - **Proactivity** (same as with Established product companies)

     Importance: 9/10

   - **English level** (intermediate high or higher)

     Importance: 7/10

- **Staffing speed** (here they need developer teams for "yesterday")

Importance: 10/10

- **Trust** (while trust is important, they could tolerate lower level of trust because they usually have very urgent needs)

Importance: 6/10

Given these problems, we could fill part of Competency Matrix:

	Established product companies	Small consulting companies	Mature stage startups	Problems
	10	8	9	Proactivity
	8	6	7	English level
	7	8	7	Development expertise
	9	2	5	Stability
	6	8	10	Staffing speed
Importance of each problem to each customer	10	9	6	Trust
	5	10	5	High-flexibility

Identifying importance of solutions to customers

Given above mentioned problems our company offers following solutions:

1. Problem: **Proactivity**

   Solution: we solve this by providing really proactive people in dedicated teams. Proactivity is one of our 5 values, and we achieve it by selecting people first by proactive attitude on interview stage and if person is not proactive - we doesn't

hire that person even if he/she is skilled enough. Detailed definition of "proactiveness" and method by which we select such people is beyond the scope of this report. After we selected people, we provide trainings for people so that they understand more clearly what we mean to be proactive and how this culture could be spread to other team members.

2. Problem: **English level**

   Solution: we solve this by providing people with Intermediate high level of English (or higher). Even if level of some person is lower, we include that into performance improvement plan for this person and compensate by providing Project Coordinator who makes sure that communication of this person is acceptable and is not a bottleneck in engineering process.

3. Problem: **Development expertise**

   Solution: all developers pass career path theory and practice skill assessments including all necessary technologies and skill levels. If some specific knowledge or expertise is needed for certain client - we include this into performance improvement plan for the engineers working with this client and ensure that engineers improves it's knowledge over time. If this is not happening and it becomes bottleneck - we proactively look for another engineer that will better suit the team (this is direct responsibility of assigned Project Coordinator)

4. Problem: **Stability**

   Solution: we assign one engineer to one project at a time and each engineer has assigned HR person who tracks satisfaction level of engineers. Before onboarding people on projects where stability is important we speak with HRs and engineers to make sure that they are satisfied and will stay on the project for a minimum of one year. If we don't get clear commitment from engineer backed by HRs note - we are not assigning this engineer to project

5. Problem: **Staffing speed**

Solution: we have separate Recruitment department whose sole responsibility is to increase staffing speed while having the same level of engineering skill level. Also, responsibility of our Project Coordinator is to know 3-6 months plans of client in terms of projects so we often open vacancy 1 month before there's actual request from client, so when client makes request - we already have several candidates to interview. Besides this we have engineers that end current projects and become available for another projects and in this case we immediately look if this is good match for client with request.

6. Problem: **Trust**

    Solution: this is being solved by multiple factors:

    a. Maintaining high social proof rating on review platforms such as Clutch, GoodFirms, Upwork and others (this is achieved by doing our best to deliver expected services to our clients)

    b. Providing ability to speak with every member of dedicated team

    c. Visiting clients and inviting clients to our offices for working together with the team

    d. Maintaining good and formal communication with summaries after each call in the beginning of co-operation to make sure that we understood each other correctly

7. Problem: **High-flexibility**

    Solution: we offer more flexible involvement of engineers as additional option for which we charge higher rates

Identifying competitors and alternatives

In order to identify competitors / alternatives according to Competitive Matrix there are 3 questions to be answered:

1. Who do we see in RFPs? Who do customers tell us about?

    During the last 6 months in RFPs we saw following competitors:

    a. Indian companies

    b. South-America companies

    c. Eastern Europe companies

2. Who is investing to compete now / near-term?

    a. African companies

    b. Chinese companies

3. Who could compete, but isn't competing today?

    a. AI-powered software development practices

While we keep an eye on long-term risks of competition, out of these I would identify only 3 competitors for the sake of narrowing down the analysis:

1. Indian companies
2. Eastern European companies
3. South-American companies

Given these options and what our prospects told us (see Appendix A "Client research information") we can't compete with Indian companies by price, but we could compete using Proactivity level and we should use this as our differentiator. Also, we could compete with Indian companies using Trust as we provide very transparent co-operation with our clients and this is proven by Social Proof of our company on the web. And pretty much the same goes for South-American and Eastern-European companies' comparison. Though market is saturated with competitors, market needs are growing 30% a year so it makes sense to stay in this market investing into competitive jockeying and differentiating options. In order to stay competitive, we need to go from competitive

jockeying on monthly rate to differentiators and we identify following differentiators as priorities for us in near term (1 year):

1. Proactivity of dedicated teams - differentiate
2. English level - stay competitive (competitive jockeying)
3. Engineering expertise - stay competitive (competitive jockeying)
4. Stability - stay competitive (competitive jockeying)

In long-term (1-3 years) we need to differentiate on following things:

1. Proactivity of dedicated teams - involve Business Analysts, UX designers, Software Architects to provide more proactive solutions to client's problems rather than just dedicated teams
2. Engineering expertise - differentiate using additional capabilities like Machine Learning, Blockchain, IoT and others

Delivery department structure

Following diagram demonstrates responsibilities of CTO at Sombra (Head of Delivery role diagram will follow):
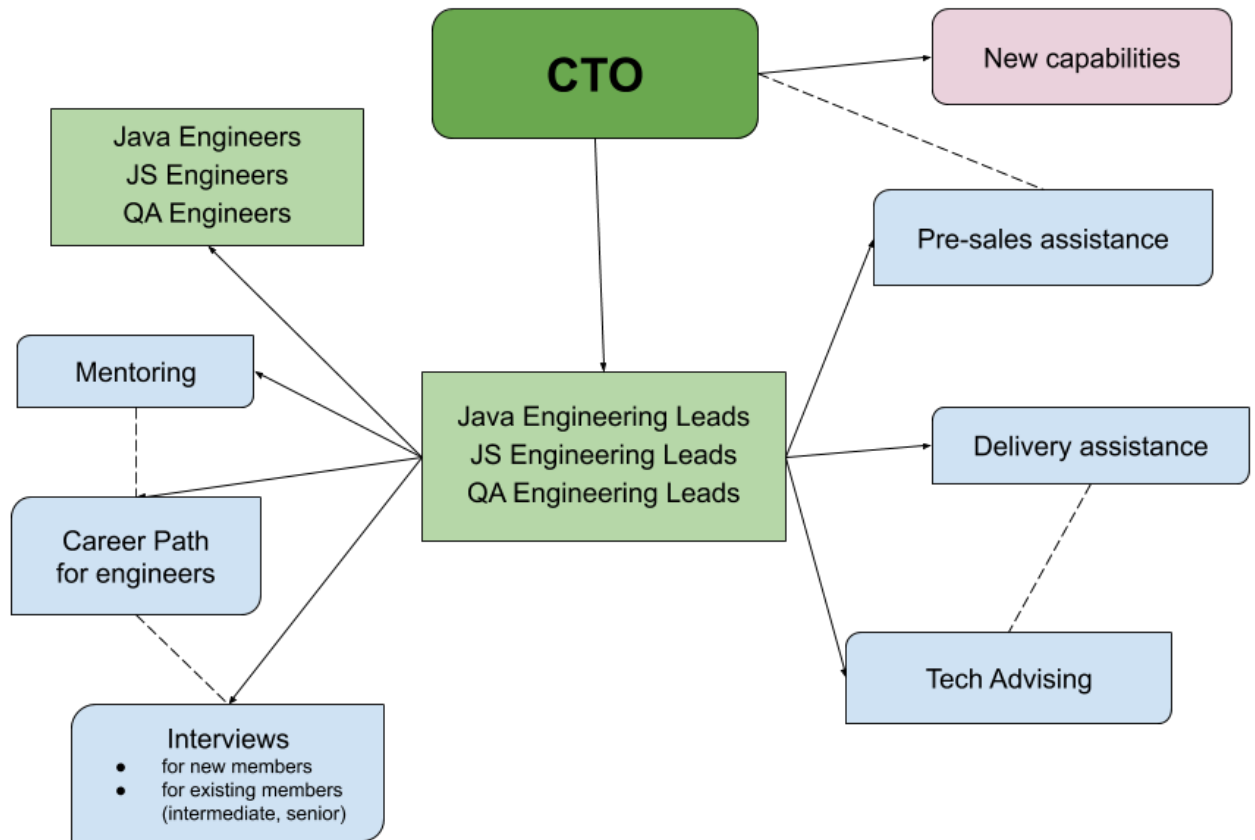
CTO position at Sombra currently performs 2 roles:

1. Actual **CTO role** where he forms vision of what company sells from technology perspective in near- and long-term. This is done by working closely with sales team, understanding current market and trends, understanding competitor offerings and planning what next capabilities should we evolve

2. **Head of Engineering role** where he oversees pool of engineers, maintains existing capabilities at necessary level and develops new capabilities following requests from CTO role

On the other side of delivery department there is Head of Delivery position who is responsible for delivery of the projects to clients and oversees that all client's expectations are met - whether scope / time / budget expectations or dedicated team expectations. Please refer to the following diagram to better understand separation of responsibilities between CTO and Head of Delivery positions at Sombra:
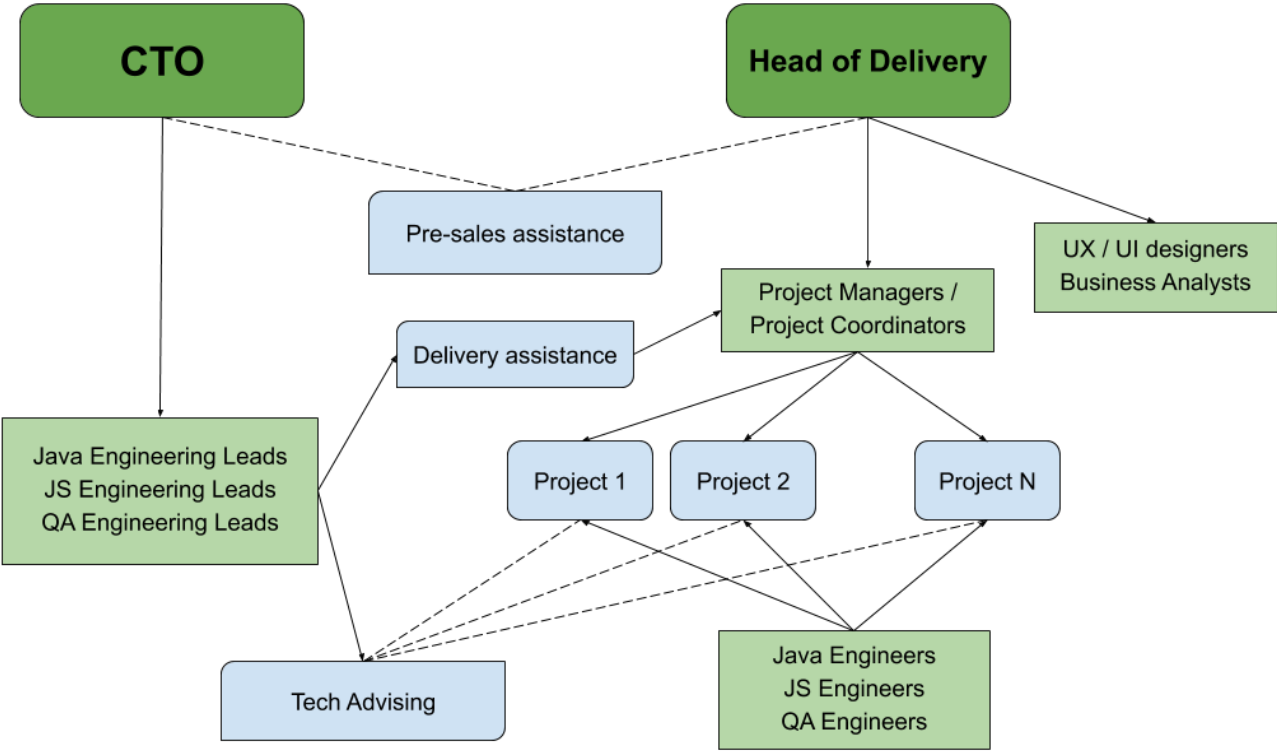
Let's start from describing engineering pools and then proceed to Project Management / Project Coordinator pools as well as UX/UI designers and Business Analysts pools.

Currently Sombra has 3 engineering pools, these are:

1. Pool of Java engineers enclosing following capabilities:

    a. Development & support of backends for SaaS and other systems

    b. Development & support of databases (both SQL and noSQL ones)

    c. Application cloud infrastructure support (deployment of above mentioned applications to cloud providers such AWS, GCP, Azure and other cloud providers)

    d. Software Architecture capability identifying non-functional requirements and building architecture for developing solutions following NFRs

2. Pool of JS engineers enclosing following capabilities:

    a. Development & support of web frontends for SaaSs and other systems

    b. Development & support of desktop frontends

    c. Development & support of hybrid mobile frontends (Ionic, React.Native)

    d. Development & support of backends using Node.js stack

    e. Development & support of databases

    f. Software Architecture capability

3. Pool of QA engineers

    a. Requirements management & manual testing capability

    b. Automated functional testing capability (Selenium, Protractor)

    c. Automated API testing capability

    d. Load & stress testing capability

Each of these pools have designated leads that are responsible for other activities of delivery department such as:

1. Interviewing

This is essential part of all engineering process because engineering pool starts from engineers of selected capability. They are selected by rigorous recruiting process big part of which is technical interview performed by Capability Leads to make sure that only skilled engineers are being hired and that they'll bring value. So, one of responsibilities of Capability Leads is to interview potential candidates for projects Sombra is working on. Interviews are being carried out against defined list of skills for each position (like Middle Java engineer, Senior Frontend engineer) to minimise subjective assessments of people skills. This directly influences clients because clients usually want smartest people to work for them

2. Career path development

Above mentioned defined list of skills for each position is part of career path for each position. So, first part of career path of Frontend engineer looks like the following: Trainee Frontend engineer -> Junior Frontend engineer -> Strong Junior Frontend engineer -> Middle Frontend engineer -> Strong Middle Frontend engineer -> Senior Frontend engineer -> and so on. During last 5 years of operation and working with different clients we carved out common expectations of most clients from skill levels and created defined list of these. This ensures 2 things:

   1. That engineers estimated as Middle level engineers will usually be perceived like that by clients

   2. That each engineer has career path and knows where to grow. And this career path complies with client expectations

Current career path is not exactly the same for all engineers of the same direction and allows several placeholders, which are added based on client needs. So for example if engineer lacks certain expertise with AWS, but current client needs it - it's included into career path for next level even though it's not in standard career path

3. Mentoring

Each engineer has assigned mentor from the first day and capability engineers have CTO as their mentor. Mentors sole responsibility is to help their mentees progress through career path as effectively as possible - they provide advice, help with day-to-day questions and accomodation and act as a guide

4. Pre-sales assistance

When doing pre-sales of a new project, all technical needs of client are being gathered and sent to CTO or Engineering leads to be analyzed and best solution to be provided. Because currently there aren't much engineering leads, they all know each other so it's easy to inquire among them and find out who could be the most helpful person for certain potential client

5. Tech advising

On projects that are developed by us as full-cycle solution and where we are responsible for architecture there's a separate role assigned whose sole responsibility is to make architectural reviews of the project and provide feedback to Project Manager about potential implications and architectural debt. Also, when team couldn't agree on architectural decisions it may call for advice from tech advisor. This is better than assigning architect role to tech lead on the project because often tech lead may have locked-in view of the project and not see issues that would be obvious to reviewer that is not from existing project. Such reviews lead to productive discussions between tech advisor, project manager and project tech leads which in turn lead to the best solutions. It's important to note that tech advisor doesn't carry any responsibility apart from just providing regular architectural reviews. Engineers with these roles are usually most experienced ones with many years of diverse experience on different projects

6. Delivery assistance

On projects where we aren't responsible for the architecture there still may occur situations where existing team may face challenges and need advice from someone

not from existing project. This usually arises on projects where there are only backend engineers and at some point, occurs part of frontend work. While usually project engineers implement new desired functionality by themselves, they often may need an advice and the most effective way for this is to involve most senior people whom are engineering leads

So, from organizational structure point of view CTO currently performs more cross-functional overview of projects ensuring that sufficient level of quality is delivered to each project.

On the other side, Head of Delivery is responsible for project-by-project work, standard triangle of scope-time-budget and meeting other client expectations that were passed to him from pre-sales project or discovered by him during co-operation with client.

Head of Delivery also oversees pool of UX / UI designers and Business Analysts in the similar way as it's done with engineers, however because number of these people is currently less than 10, it's totally sufficient to lead them on person-by-person basis rather than developing some strict process for this.

Sales department structure and numbers

By the end of 2017 top management of Sombra understood that having Upwork as major sales channel bringing 90% of leads is very risky and decided to diversify sales channels. Following sales channels were established by July 2017:

1. Current customer - new projects from same customers
2. References - new clients referenced by our existing customers
3. Email campaigns - leads obtained by sending cold emails
4. Linkedin - leads obtained by connecting with potential clients on Linkedin and sending cold messages there / building a network there
5. Inbound - leads coming from our website, Clutch, GoodFirms or other public profiles on the web
6. Personal networks - leads coming from personal networks of CEO, CSO, CTO and other employees
7. Business trips - leads coming by meeting portential clients in person in their cities or on events
8. Sales partners - leads obtained from different sales partners
9. Upwork

Lead coming from any of these channels was assigned to Engagement Manager and entered presales phase that consists of following stages:

1. Discovery - lead has just entered pipeline with minimun information from channel (usually text information + some documents)
2. Qualification - contacting the lead and asking all relevant questions to find out whether certain lead is a fit for us
3. Proposal - preparing proposal for the lead and sending it

4. Negotiation - settling details of our future co-operation
5. Closed - lead won or lead lost

All information about lead was entered by Engagement Manager into Pipedrive CRM. Following diagram is provided to better explain presales process:
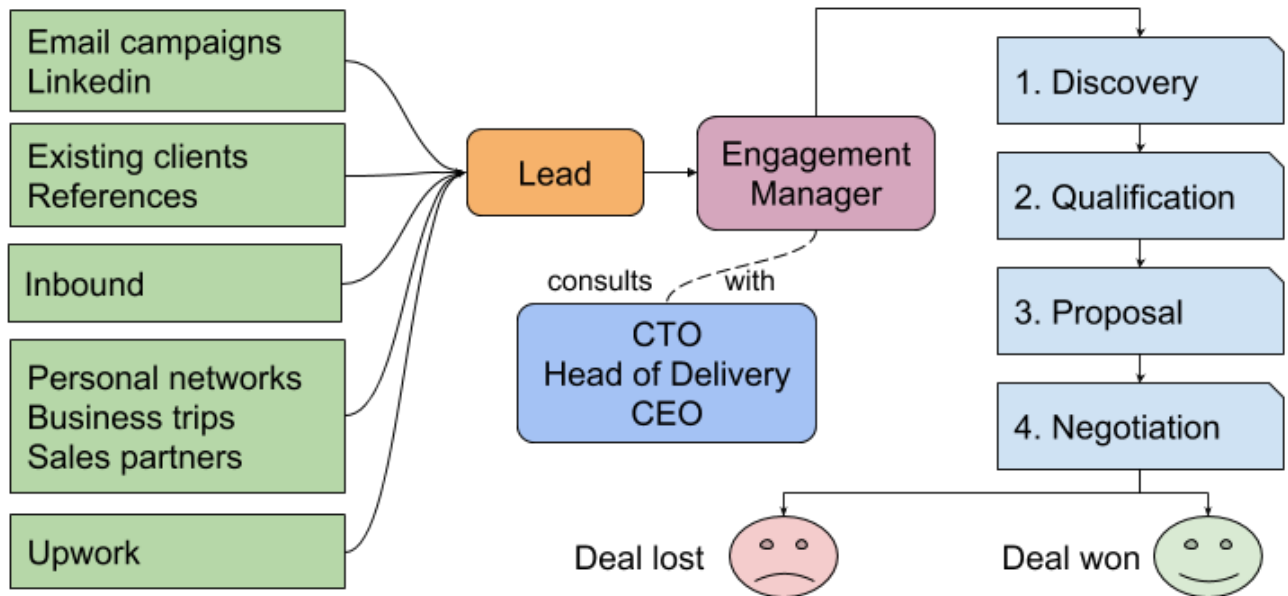
So the each lead is assigned to Engagement Manager, who reads information about it in discovery phase. Then manager proceeds to contacting the lead with the goal of better understanding client's needs. At these and following stages manager could involve CTO, Delivery Director or CEO to help him/her.

All information should be entered into Pipedrive CRM for later usage.

If deal is won - it is being handled to delivery, otherwise it is marked as lost.

Starting from October 2018 top management started to hold regular weekly meetings with Engagement Managers to keep track of what is going on during the stages and each lead was given small amount of attention.

Solving revenue problem

Increasing number of leads

By the end of July 2017 sales team managed to reduce number of leads coming from Upwork from 90% to 60%, so general percentage distribution across channels looked like (excerpt from Pipedrive CRM):

1. By number of leads:

Current customer	2%
Email	1%
Inbound	10%
LinkedIn	1%
Personal network	8%
Reference	4%
Sales partner	7%
Upwork	59%

2. By total amount of revenue:

Current customer	6%
Email	0%
Inbound	48%
LinkedIn	1%
Personal network	13%
Reference	2%
Sales partner	3%
Upwork	22%

Business Trip	4%

3. By amount of revenue / lead:

Current customer	23%
Email	2%
Inbound	37%
LinkedIn	6%
Personal network	13%
Reference	3%
Sales partner	4%
Upwork	6%
Business Trip	6%

These statistics have shown us that in terms of effectiveness (revenue / lead) Upwork is definitely not the best sales channel so it was decided to expand other channels further. We started investing more time into expanding personal networks through Linkedin, business trips, more intensive inbound strategies and more intensive work with sales partners and our existing customers.

This led to following sales statitics for Q1 of 2019 (excerpt from Pipedrive CRM):

1. By number of leads

Current customer	7%
Email	0%
Inbound	18%
LinkedIn	10%
Personal network	12%
Reference	1%

Sales partner	4%
Upwork	37%
Business Trip	10%

2. By total amount of revenue

Current customer	8%
Email	0%
Inbound	22%
LinkedIn	27%
Personal network	2%
Reference	0%
Sales partner	13%
Upwork	18%
Business Trip	9%

3. By amount of revenue / lead

Current customer	11%
Email	0%
Inbound	12%
LinkedIn	25%
Personal network	2%
Reference	0%
Sales partner	36%
Upwork	5%
Business Trip	9%

Close attention to presales process in Q1 2019

At the same time, because new sales were critcal for our company in Q1 of 2019 it was decided that top management will pay more attention to presales phase by examining all bigger leads coming through the pipeline. Following are statistics on sales pipeline of Q3 and Q4 of 2018 (excerpt from Pipedrive CRM):

Stages	Current customer	Email	Inbound	LinkedIn	Personal network	Reference	Sales partner	Upwork	Business Trip	Total Leads	
Discovery	3	2	14	2	11	6	9	82	9	**138**	
Qualification	1	0	5	0	4	0	3	9	1	**23**	**16.67 %**
Proposal	0	0	1	0	3	1	1	3	3	**12**	52.17 %
Negotiation	1	0	1	0	1	0	1	2	2	**8**	66.67 %
Closed	0	1	0	1	0	0	0	2	1	**5**	62.50 %
										Closed / discovery	**3.62%**

After starting to pay more attention by top management in Q1 2019 this sales pipeline statistics changed to following:

Stages	Current customer	Email	Inbound	LinkedIn	Personal network	Reference	Sales partner	Upwork	Business Trip	Total Leads	
Discovery	6	0	16	9	11	1	4	33	9	**89**	
Qualification	2	0	9	3	3	0	2	9	2	**30**	**33.71 %**
Proposal	2	0	3	1	1	0	1	3	2	13	43.33 %
Negotiation	0	0	2	2	1	0	0	3	1	9	69.23 %
Closed	1	0	2	2	0	0	0	1	0	6	66.67 %
										Closed / discovery	**6.74%**

This has shown that after looking more thoroughly at presales process and assisting people involved into lead engagement process, conversion from discovery leads increased almost twice - from 3.62% to 6.74%.

This proved that besides increasing number of leads it also makes sense to make engagement process more efficient.

Presales process issues

During taking part in many presales activities processing leads during Q1 2019 it was noted that prevalent majority of these included substantial amount of complex technical information and sometimes non-technical engagement managers had troubles understanding it in the correct way. Other issues included the fact that clients didn't

described their needs in the words that we use so there were a lot of misunderstanding during presales phase as well.

Issues encountered during different stages of different leads included:

1. Asking non-technical people (CEO, COO, Product Managers) technical questions
2. Presenting very technical details to non-technical people
3. Not involving technical people on the calls with technical people from client side
4. Presenting very fine-grained details to people that were interested in general numbers
5. Spending much time on requests non-decision-makers
6. Not asking budget / timeline expectations before making proposal
7. Using wrong communication channels (emails instead of videocalls)
8. Proposal not consistent with needs of client
9. *(and many others)*

These issues happened randomly in different people involved into sales process and sometimes we had troubles aligning expectations about same lead on weekly Engagement meetings.

Updated presales process

Mentioned above inconsistencies were indicator of the fact that presales process were very dependent on certain Engagement Manager. This led to idea of making this process more predictable and universal. Because the most inconsistencies turned out at Qualification and Proposal stages of presales process, updated process looks like the following:
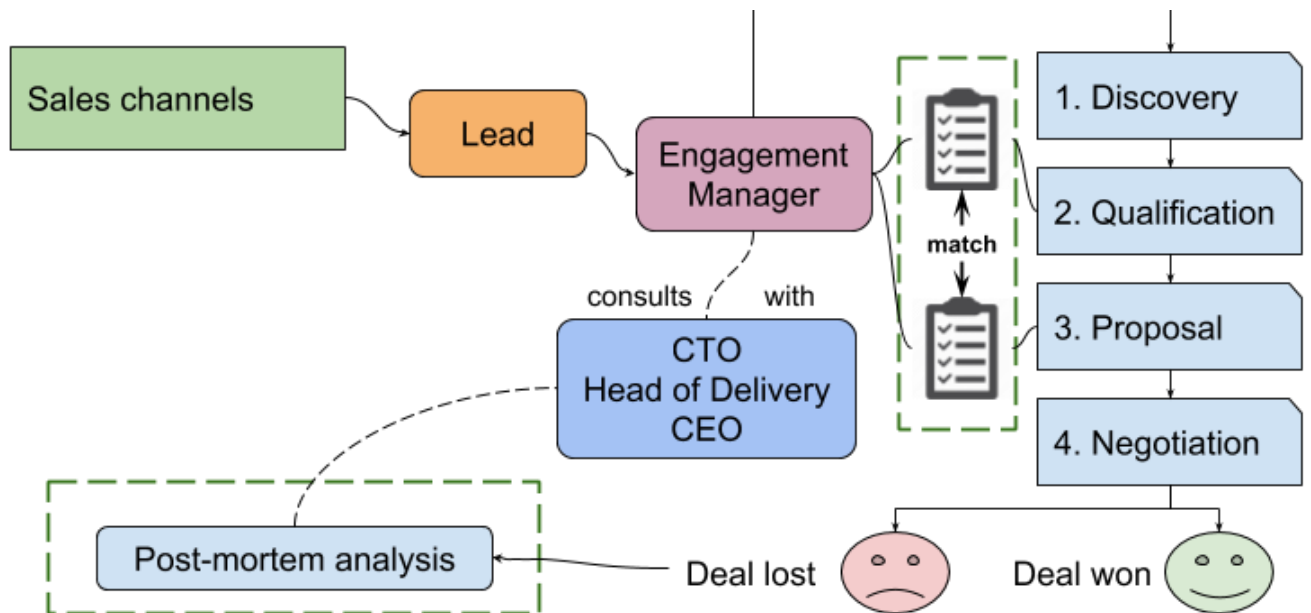
1. Get all available information from the web and information shared by the lead
2. Fill in Qualification stage checklist (see next chapters)
3. Initiate meeting / videocall / call to ask relevant questions to finish Qualification stage checklist. At this stage it makes sense to involve technical person - CTO, Delivery Director or other one
4. Once information is there, engagement manager could start preparing proposal for client filling Proposal checklist
5. Cross-check Proposal checklist with Qualification checklist to not miss any information from the latter one
6. Rest of the process stays the same if deal is won

7. If deal is lost - post-mortem analysis should be performed to see if we could've done anything other to win this deal. This step is especially important, because lost deals provide the most information to tailor value proposition

Qualification stage checklist

Engagement manager should gather all available information about the lead from public web, initial description of needs / requirements and then proceed to the call with clear set of questions to understand client and needs in the most detailed way possible.

Hence it is advised to follow this checklist:

1. **Business context -** company / venture information
    1. Type of business (see chapter "Identifying client groups")
        1. Early stage startup
        Could be identified by the fact that they haven't built MVP and only plan to approach investors for the first time
        2. Mature startup
        Could be identified by the fact that they have attracted first or subsequent rounds of investments, but rely 70% or more on these investments (in other words, without investments they wouldn't survive a year on their own)
        3. Established product company
        Could be identified by the fact that they have software product which they sell to their clients and this is profitable business model i.e. they rely 80% on payments from their customers. They may attract investments or take loans however they won't be higher than 20% of planned yearly revenue
        4. Small consulting company

Could be identified by the fact that they provide consulting to bigger clients and doesn't rely on investments (similar to Established product company)

5. Large enterprise

   Large enterprise with diverse set of different products that they sell and company history

6. Other?

2. Size of business in terms of people / yearly revenue

   Most owners are very open to share this information and this item is very important to understand size and structure of business. From this one could understand higher amount of revenue that this client could pay us (e.g. if expected yearly revenue of company is 120,000$ then this means that it doesn't make sense for customer to pay us 10,000$ for development).

   On the contrary if company has 30 full-time US employees working in the office, then this means that company has revenues that covers salaries and office rent in US.

3. Their customers

   1. B2B or B2C?

      These are totally different business models, and this puts different implications on business structure and yearly goals

   2. Key customers / customer segments?

      Most companies will tell who their key customers are and what are their yearly goals towards those customers

   3. How many customers?

      Number of customers also means maturity of the company or risks. If company has only one major customer - this could be great risk from the long-term perspective, and it makes sense to discuss that in details to

understand if long-term co-operation with this company is mutually beneficial

4. How they make money / plan to make money?

   This is more important question for startups, especially early stage startups to understand their vision of the business

5. Goals for next quarter / end of year:

   1. Acquire new customers
   2. Expand co-operation with existing ones
   3. Merge with another company
   4. Buy another company
   5. Be acquired
   6. Sell product to another company
   7. Other

   This is very important, because co-operation with us as a company fits into one of such goals and this is potential to do more for our client - be more proactive and help them achieve their goal instead of just providing dedicated team

6. Where is software there and where we fit there?

   If client's main goals don't include software and us as a company, this means that we're not important for the client and we could be stripped off at any time - this is a risk which should be assessed

7. Org structure related to decision-making with software

   Usually decisions are made by CTO or VP of Engineering, however it could get more complex than that and it's very important to know who's making the decision. We have a client where CTO is not related to work with vendors, while Engineering lead does

2. **Personal context** – whom I'm talking to?

1. Position in the company and responsibilities
    1. CEO
    2. CTO / VP of Engineering
    3. COO
    4. Product / Project Manager
    5. Software Architect / Technical Lead
    6. Other

This is very important to know because for example CEOs usually think high-level and delegate details to their team (CTO, Head of Finance etc), so it could be a good idea to help them delegate different questions instead of just ask those questions directly.

2. Personal plans in this company
    1. Owner and plans to evolve this business
    2. Owner and plans to sell this business
    3. CTO that plans to leave this company in 6 months
    4. Other?

If we are discussing 2+ year co-operation with decision-maker who plans to leave the company in 6 months, then this is a great risk which should be discussed in details

3. Sales lead type
    1. Gatekeeper
    2. Influencer
    3. Decision maker
    4. Self-proclaimed decision maker
    5. Blocker

This is usual sales workflow to understand to be more effective and helpful during presales phase

4. Level of technical knowledge
    1. Very technical

2. Somewhat technical

3. Non-technical with some technology knowledge

4. Totally non-technical

5. Non-technical, but claims to be (identifiable by using a lot of buzzwords like AI or blockchain without clear vision on how they should solve the problem)

Technical people should talk with technical people and if it's not like that - it could be great risk to presale process. On the contrary, non-technical people doesn't like to listen to very technical details so Engagement Manager should understand whether amount of technical details presented is just about right

5. What is the goal of speaking with me / us as a company?

There were cases where people claim to have a need but don't have it actually. This is absolutely ok however it makes sense to know it before spending too much time on preparing proposal that is not needed

6. Business interests - what person is interested in terms of business (for example person may be interested in business volunteering, charity etc)

This could be very useful is we pursuit same business interests because this could increase mutual trust

7. Basic personality information

1. Likes to talk much vs very laconic

2. What triggers him/her to talk more?

3. **Their needs / problems** to be solved

1. Pure staffing needs

1. Expertise

1. Single stack (Java, JS, .NET, Python, Ruby, Go)

2. Mixed stack (Node.JS + Angular, Java + Angular, Python + React?)

2. Expected size of team in 6, 12 months if everything goes well

3. Existing technical team and org structure

1. CTO, Head of Engineering, Product Manager, Project Manager

2. Inhouse team and structure

3. Inshore / offshore team(s) and previous expertise

4. What is the problem they are trying to solve?

   We had multiple cases when non-technical people thought that they need mobile app and were desperately looking for team of native mobile developers when it turned out that they simply need responsive mobile website. Giving simple piece of advice increased trust and led to good relationships with these people

4. Business trips needed

   As mentioned in chapter "Product management perspective", this is very important part which increases understanding and trust significantly and very often it's not perceived as additional expenses from client's perspective

2. Turnkey solution

   1. UX wireframes / UI designs

      In order to increase efficiency of delivery it makes sense to invest time and money in designs, because it's cheaper than modify code of already built solutions

   2. Functional requirements present or needs to be prepared

      If requirements are not written down in clear and structured way, again in order to increase efficiency of delivery it makes sense to invest time and money in requirements elicitation since this is again cheaper than rebuilding code several times after it was built

   3. Software Architecture

      1. Technical Constraints (e.g. must be deployed to IBM Cloud)

         Did we take into account all technical constraints? Were these constraints clearly agreed with another party? If not - this is a great risk that some

additional technical constraint may pop up right before the release and require a lot of effort to be closed

2. Business Constraints (e.g. built solution to be deployed by our DevOps team only)

   Same as with technical constraints, business constraints like the need for the project to be managed by their team or the like may significantly influence architecture

3. Non-functional requirements

   These are defining requirements for the architecture and they are key architectural drivers which influence architecture. These are often overlooked for the sake of complete functional requirements. These must be reviewed and if needed elicited by software architect and also clearly written down with measurable outcomes

4. Date of first release, reason for first release (demo), number of users

   While this should be discussed during requirements elicitation phase, there were cases where simple reasons of first release (like demo to existing customer with specific demo data imported into the system and thoroughly tested before the demo) were not clearly written down which lead to mismatched expectations, not satisfied customer and lost opportunities

4. **Timeline / budget expectations**

   It's important to ask this question several times especially in dedicated team contracts because often customer tell that they need just a team for 3 months that they'll manage on their own. However, after 2.5 months they appear with complaints that deadlines weren't met because of the team and demand compensation. These cases are usually well covered by MSA, SLAs and SoWs however it's more about understanding and satisfying real customer expectations than working out legal issues.

5. **Communication channel and timing**

   1. Ensure best communication channel possible

      1. Meeting in person is the best and most preferable way of communication

      2. Videocall is worse that meeting in person, however still acceptable

      3. Audiocall without video should be avoided at all costs

      4. Instant messaging (Skype, Whatsapp, Slack) - good for discussing quick details (like date of start), however very bad and time consuming for negotiating a lot of different details

      5. Email with summary should follow after each meeting

   2. Is this person interested in replying to you?

      Good question to ask yourself before you send message to person - are you sending useful information to this person and will this person be interested to reply? Or it is just another "for your information" email that doesn't interest person to reply?

   3. When is our next communication in terms of time?

      Clear agreement of next steps and dates of next communication and reply

   4. What this person would like to see / hear next time?

      Clear agreement of what is next thing this person should see

      1. Short IM message

      2. Invitation to next call

      3. Long email summary with details

      4. Answers to discussed questions

      5. Short standard presentation

      6. Long customized presentation

      7. Video about company

      8. References from clients

      9. Public reviews about the company

10.Other?

5. Content format

   1. What type of content (email, presentation, presentation + call, official proposal)

   2. How much time this person will spend reading your message?

This is especially useful to know to save time when preparing answer. If person expects to see email with several sentences, it doesn't make sense to conduct major research to write those several sentences

6. **Financial / Legal details**

   1. Which documents should be sent and signed (NDA, other?)

   2. Legal details?

Financial and legal documents could often be a stopper for lot of contracts and while we are flexible it still makes sense to plan time ahead for this type of activity

7. **Other concerns**

   1. Security

   2. Where our developers are located

   3. Certifications (ISO9001, PCI-DSS, HIPAA)

   4. GDPR

These usually arise during requirements elicitation phase, however if customer repeats them often - it's his pain point and it makes sense to re-assure him/her again that these specific needs will be addressed.

Proposal stage checklist

1. **Who will see proposal?**
    1. How much time this person will spend reading your message?
    2. What does this person would like to see / hear?

2. **Proposal content format**
    1. What type of content (email, presentation, presentation + call, official proposal)

3. **Presentation / document**
    1. Title
    2. Agenda
    3. How we understood your problems / needs
    4. How we plan to solve it
    5. Why we believe our solution will work
        1. Our differentiators list (differs for each lead)
            1. References from relevant clients from this area (US, UK)
            2. Sombra office visit, customer office visit
            3. Dedicated team
                1. Turnover rate
                2. C-level attention etc
                3. Customer success stories and references
                4. Access to big pool of talent & rigid recruiting process
                5. Values (Customer, pro-activity, honesty, self-development, changes)
            4. Turnkey solution
                1. Solid architecture experience (solid understanding of FRs, NFRs, Constraints)
                2. Solid engineering team (certificates)

3. Solid delivery experience (solid understanding of budget / time / scope limitations and honesty)

4. Our values (proactivity, honesty, customer, changes, self-development)

Financial aspect

From financial perspective new presales process means more involvement of CTO and Head of Delivery to:

1. Train Engagement Managers to follow new presales process and control it

    This means that either CTO or Head of Delivery will consult Engagement Managers during first 3 months during presales work on new deals. Given the number of deals for Q1 2019 (see chapter "Close attention to presales process in Q1 2019" of this diploma):

    a. Qualification phase - 30 deals

    Each deal at this phase takes at most 15 minutes of CTO or Head of Delivery time to give advice to Engagement Manager (5 minutes to understand context, 5 minutes to give advice and 5 minutes to switch back to previous task), amount of dedicated time will be ~ 450 minutes = 7.5 hours

    b. Proposal phase - 13 deals

    Each deal at this phase takes at most 30 minutes of CTO or Head of Delivery time to give advice to Engagement Manager (5 minutes to understand the context, 15 minutes to give advices and 10 minutes to switch back to previous task), amount of dedicated time will be ~ 450 minutes = 7.5 hours

2. Perform post-mortem analysis of lost deals

    Post-mortem analysis usually performed for deals that didn't pass Proposal stage and given that this number won't be higher than 15 and time to do post-mortem is 20 minutes, amound of dedicated time will be ~ 300 minutes = 5 hours

So, in total this process will take 20 additional hours of CTO and Head of Delivery, which we will calculate based on hourly rate of 100$ / hour (which is about 2 times higher than actual one). So new presales process means spending additional 2000$ / month worth of CTO / Head of Delivery attention to presales process.

Because presales process belongs to income generating activities, this means that assuming that:

1. New deal brings 10000$ / month
2. Expected Closed / Discovery ratio grows to 9% (now 6.74%) - this is the reason of presale process
3. In Q2 we'll receive 100 discovery leads, which will convert to 9 Closed deals / Q2

So estimated revenues from new presales process should generate 3 deals / month worth of 10000$ / month / deal = 30000$ / month which is significantly higher than invested time of CTO / Head of Delivery.

This also allows CTO and Head of Delivery to be closer to new clients and market demands meaning that they'll notice changes in trends faster and be able to adjust accordingly.

This proves that financially speaking, new presale process which involves CTO and Head of Delivery does make sense.

Executive Summary

1. Why this project was chosen as diploma work?
   The reason why this project was chosen is because this was the major problem that company faced in the end of 2018. It started from lower sales figures, then

significantly impacted our financial results by shrinking profits, then expanded to HR department because bench started to grow, and some people started leaving company because they haven't been involved into projects. From the theory of constraints approach, sales became the bottleneck and our managerial decision was to focus attention of the most experienced people on the sales process. It was understood very clearly that other departments and activities won't make sense during next year (2019) if there are not enough sales. While closely working on sales, we found 2 problems:

    a. Lack of qualified leads

    b. We were able to win only very hot and "desperate" leads outside of Upwork

This was caused by the fact that we stopped relying on Upwork as main sales channel and very soon understood we don't know how to find and filter potential clients outside of Upwork and also how to identify their needs in a fast way. This all happened because for 5 years we heavily relied on Upwork, which provided all needed information on the plate:

    a. Huge number of new leads added daily

    b. Most leads with clearly described and usually hot need (ready to start in 1 - 4 weeks)

    c. Previous co-operation history, reviews and rates suggestion

This didn't come for free and Upwork had other drawbacks:

    a. Low rates: Upwork is very often seen as very cheap option and "last resort" for people / companies. So, it's very rare that companies that are ready to pay higher rates get to Upwork, because they usually get better offers by companies operating in their local cities / countries

    b. Huge competition: each newly created lead is bombarded with 20 - 50 competitor proposals during the first day and most of them are from countries with lower rates

Despite these, we were able to remain highly competitive and grow the company from 3 to 120 people in 5 years, but at the price of not really developing other sales activities and taking our growth for granted without putting much effort into it. So in the end of 2018 as a CTO that worked before as a software developer in several product companies and understanding their needs it was clear that I may significantly improve presales process by using knowledge obtained from Lviv Business School courses and applying it on a day-to-day basis to pre-sales cases.

2. What are the main achievements of this diploma in terms of company?

Besides financial results and new clients obtained during pre-sales process there were a lot of knowledge and experience generated:

a. Understanding of client needs

During this period, I had a chance to follow almost whole cycle of standard sales process - from prospecting to working with cold leads to making them warm and taking part in closing process. This brought great understanding of the market where we are operating and so diverse segments of client needs in the "so called" out staffing market, which is claimed to be not very technical from the complexity of solutions that are delivered. For instance, there is whole segment of clients that just needs dedicated engineers and doesn't care if they are in Ukraine and sitting in one office or scattered around the world. And as a result, I met competitors who built working business models on just re-selling engineers from other companies. This led me to better understand that one of our differentiators and valuable assets are engineers that we worked with for several years and we are completely confident in their abilities.

b. Understanding of market needs

Speaking with different people in London and US it's clear that there's global shortage of skilled engineers in the world. Big companies like

Google, Facebook take the most skilled engineers from US market of engineers, big banks and financial corporations do the same on UK market. This leads to vacuum in terms of need of skilled engineers just doing the work right and using modern tools. Of course, along with huge demand there's huge supply of engineers from companies like ours. However, after speaking with our existing and new clients - they confirmed that we offer best engineers they had a chance to work with among other companies and they will reference us for sure. Taking into account higher hourly rates we started working with after Upwork this gives some fresh air in terms of profits and ability to involve even more senior people to bring more value to our clients.

c. Understanding of competitors

During meetings with prospects it turned out that some of them already had vendor they were working with for several years and were quite happy with and they were

While speaking with potential clients I was also approached both in person and on Linkedin with various companies that offered similar services to me and this was also a good experience understanding what and how they sell it.

3. What are negative sides of this diploma in terms of company?

During the Upwork "era" at Sombra I usually focused on working tightly with software engineers and project managers consulting them on their projects, creating career paths of engineers and helping them to grow in our company and bring more value to our existing clients. Since October 2018 and until April 2019 I dedicated most of my time to sales activities and thus reacted only to the most critical cases raised by Delivery department. This delayed my plans to grow expertise and gravitate towards outsourcing-based delivery rather than outstaffing-

based one. Also, some of key engineers felt this departure and started questioning my involvement and technological focus that company is pursuing.

4. What are next steps?

   a. First step would be to sustain new pre-sales process and produce small victories according to Kotter's 8 Steps and make sure that pre-sales process doesn't rolls back to just working with hottest leads and skipping more warmer leads.

   b. Second step would be to return to closer work with key engineers and project managers to consult and bring value to existing clients and extend co-operation with them.

   c. Third step would be to grow expertise and learn to sell outsourcing projects and solutions by looking at the companies in US / UK that does the same and try to sell similar outsourcing projects / solutions like them. This is more challenging from multiple perspectives: 1. Finding right companies to sell to 2. Selling outsourcing projects / solutions 3. Business analysis and requirements 4. Implementation with scope-budget-time triangle 5. Delivery and end-user training. That's why it is considered by our company as long-term investment rather than something that could bring short-term results without upfront investments

List of sources and literature used

1. New Sales. Simplified.: The Essential Handbook for Prospecting and New Business Development Paperback – September 4, 2012 by Mike Weinberg
2. Everything is Negotiable: 4th Edition by Gavin Kennedy
3. Software Architecture in Practice: Software Architect Practice (SEI Series in Software Engineering) 3rd Edition by Len Bass, Paul Clements, Rick Kazman
4. LvBS courses
   a. Management Decision Making Toolbox by Mychailo Wynnyckyj
   b. Strategic Marketing Challenges by Joe Pons
   c. IT Strategies by Alex Shegda
   d. Sales Management by Mychailo Wynnyckyj
   e. Financial Decision Making by Yuri Zayarny
   f. Software Architecture for Managers by Matthew Bass
   g. Product Management by Scott Sehlhorst
   h. Projects management by Sergii Potapov
   i. Business Strategy in the Times of Assymetric Competition by Adrian Slywotzky

Appendix A "Client research information"

Client research was performed using following methods of primary research:

1. Prospect interviews
2. Client interviews

Prospect interviews

In order to better understand needs of our potential clients, 2 business trips to London were scheduled:

1. 10 meetings with prospects in 4-8 February business trip to London
2. 32 meetings with prospects in 18-29 March business trip to London

These were mainly positioning such as Product Managers, VP of Engineering, CTOs of different companies - from smaller startups to seasoned J.P. Morgan engineering executives.

During these meetings main questions that were asked were the following:

1. Have you worked with offshore companies before?
2. If yes:
   a. Why did you decide to go for offshore software development?
   b. What would you improve?
   c. Which country did you work with?
   d. How would you select new offshore software development company if you need to offsore again?
3. If no:
   a. Have you considered working with offshore software development companies and why did you decide not to?

b. Imagine perfect offshore software development company that you would work with

All of the prospects mentioned that they completely ignore cold emails and cold linkedin messages, because they receive from 3 to 10 such messages daily.

80% of the prospects mentioned that they worked with offshore software development companies in the past or are working now. 20% mentioned that they never worked with offshore development companies but heard not very positive feedbacks from their colleagues.

60% of the prospects worked with Indian companies before and told that they lack proactivity, meaning that they need to spend a lot of time micromanaging developer teams. They contrasted that with teams from Eastern Europe which are more proactive and could manage themselves - this is a great differentiator.

When asked 12% of prospects about what they understand when we call ourselves "outsourcing / outstaffing" company and they clearly told that these words mean for them "lost control of what's happening". By outsourcing they understand that there'll be one person doing all the communication and it's impossible to speak to engineers, because they're either constantly changing or belong to different companies. So outsourcing / outstaffing means cheap last resort for them and have negative connotation. On the contrary, by prospects words - what we do should be called "remote software development teams" to differentiate from other outsourcing / outstaffing companies and provide information that control of development won't be lost.

About 70% of prospects agreed that ability for our PM or team to travel to their destination is great differentiator for them, because often in offshoring companies

developers are spread around different locations and it's hard to gather them and meet with the team.

Also, about 40% of prospects told that ability to visit our office and meet with the team is a great differentiator increasing their trust, because often offshoring companies in turn outstaff people from other companies having little control of what's going on.

It's interesting that 90% of prospects didn't mentioned high or low expertise of software engineers and that's because they measured expertise mostly by proactivity and English level which is usually significant bottleneck rather than expertise level.

90% of prospects mentioned that they turn to offshore software development because most good engineers in UK are being headhunted by big corporations and banks so there's great lack of good engineers.

100% of prospects mentioned that they'll choose their next offshore software development company by recommendations in their network. If they are approached from outside they'll either ignore or look very critically at proposals.

About 50% of prospects mentioned that social proof is very important for them meaning that they would like to be confident that other our clients are satisfied working with us. This could be obtained by providing either references from existing clients or by exposing reviews about our company.

Client interviews

In the end of April 2019 Sombra had 23 active clients, which could be categorized as follows (in brackets ids of these clients in our CRM are mentioned for reference):

1. Established product company (SZ, NS, VD, FW, PW, ML) - 6 clients
2. Mature startups (PE, BS, SPM) - 3 clients
3. Consulting companies (RO, NI, FCI, PC) - 4 clients
4. Early stage startups (BM, TA, XP, DR, PS, IF, MER, TE, PH) - 9
5. Large enterprises (SW) - 1

Client interviews were performed face-to-face with them during our visits to these clients in February 4-8, 2019, during their visits to our office in September 2018, March 2019 and over video conference calls.

Main topic of these discussions were why those clients chosen us and what are our strong points that they see as differentiator. They all mentioned following strong points:

1. Fast and prompt responses to their inquiries in the beginning, which improved their trust that we'll respond in the same manner during our co-operation
2. Fast reaction to occurring problems and solutions to them during our co-operation
3. Ability to build strong dedicated teams
4. High proactivity of engineers in teams ("we like that your engineers challenge us, suggest better solutions to our problems and don't just blindly follow our instructions")