**Future Adilas Business Model Planning – Where are we headed?**

Dustin Siegel

8.6.20

How big do we want to get?

 A: It’s not a question of want, it’s a question of need. I think we need to get bigger, not huge but big enough to have enough developers to handle bug fixes, and server issues quickly, also to provide each developer enough time and tools to implement robust testing systems. Also, to have a more robust project management and code review team. Another reason to get bigger is to be able to welcome more opportunities for niche development and broaden our clientele.

What is our time frame to get there?

 A: I honestly don’t know. I would think we should start the process now.

What do we want to do/have for our company/user culture?

 A: I love our company culture and don’t want any kind of fundamental change.

 B. The dominant user cultural trait as far as I can tell is frustration with a general understanding that Adilas is the best of the options even though it is so frustrating. I think we can change that with some strong focus on making things easy/pretty.

What are some of the things you like about what we are doing right now?

 A: I love our how helpful and friendly all of the developers are. You have all helped me learn how to do this overwhelming thing (learning Coldfusion/Adilas backend). With enormous amounts of patients, understanding and care.

 B. I love how the pressure I feel to get things done comes from a desire to help out the customers instead of internal pressure from my co-workers/company.

 C. I like the flexible work hours.

 D. I like how we are inclusive in the broader company direction.

 E. I like Steve’s attitude towards traditional bosses.

 F. I like how we are willing to broaden our clientele by trying to capture more/other markets.

 G. I like how powerful Adilas is a tool.

What are some things you would want us to change?

 A: We need a team of testers on a dedicated system that test all code thoroughly before deployment.

 B. We need a broader avenue for newer developers, (like me) to get help quickly when we are stuck. My issue with the current system is that I don’t know when I will need help until I do and then not being able to get it at that point can stop me in my tracks.

 C. We need to implement a company wide solution for unit/integration testing with extensive support and training.

 D. This is probably a stretch, but I would love to see each developer (who wants too) have a second developer that shadows/works co-operatively on projects. I think that if I had another developer (or junior developer) to work with/bounce Ideas off of/team up on projects with. We would more than double the output.

As far as structure, what do you like? Do you like the independent model or would you like it more solid?

 A: I like the independent model for most developers. I think it only makes sense since people have specialties where they are more involved with specific client types for them to be able to focus on those specific development needs. However, I think that some developers are needed in dedicated roles as well, like: Integration. Testing. Bug hunting. Look and Feel. Server. Project Management. Code Review. Etc.

Do you like the jellyfish model? (Organized with a loose, flowing feeling. Adilas is the overarching umbrella and there are a number of sub departments or subsets.)

 A: Yes, but this diagram is misleading. Some of those bubbles are tiny and others are much larger if we look at this realistically then that distinction is important to address/consider.

 B. It is also important to consider the “Powered By Adilas” opportunities in addition to the Jellyfish.

Would you want to see something different (company structure)?

 A: As mentioned above; in addition to those who want to work independently on projects, I would like to see a team of dedicated developers who do Integration. Testing. Bug hunting. Look and Feel. Server. Project Management. Code Review. Etc.

 B. I would like to see a way for questions to be routed more efficiently. For Example: I get a lot of emails from consultants asking for help finding issues that end up being miss created/allocated/handled plants that don’t have a code-based solution but more of a consultant investigation solution. My instinct at this point is to look for code bases solutions and I often waste time looking at my code trying to solve a non-code problem. It’s not their fault, but it would be helpful if there was a standardized way for people to seek help where it gets routed to those with the right knowledge to fix it more quickly. This is a tough one because maybe I am the one who knows the most about the parts I have been worked on. Hmmmm……

What products and services do you think we should focus on?

 A: I honestly don’t know but I think that the future is in online marketplaces and skill/need social connectivity. We need to look at things that are easy to monetize and have low maintenance costs.

Do we want to continue to internally fund this or seek outside investments?

 A: This is a tough one! I don’t have enough financial information available to me to answer this question.

What do we need on the management side? What do we need in leaders and managers? Traditional managers? Or more of a Board of Trustees or Advisors?

 A: Solid Project Management is important but there is no need for “Traditional Managers”. I think instead we need some people in clear roles, I.e. Integration and Testing, Server handling etc.

Do we want some sort of internal loyalty system or revenue sharing plan?

 A: Yes, Definitely. That would be a strong motivator for us to keep pushing the system forward and promoting its growth.

How do you play? What is your time commitment, scheduling, maxes and mins (hours per week/month)?

 A: I have been putting in approximately 30 hours per week. However, I am motivated to put in more time, but I am held back by my own lack of skills and I end up spending too much time trying to figure things out and researching code techniques and feel guilty about it. If I had a more immediate support structure to access when I am stuck, I would definitely clock more hours/get more done each week.

What business verticals are you interested in?

 A: I am far more interested in the coding process and development in general as opposed to a specific market. Also, I don’t have enough knowledge to know which would be profitable or not.

Where would you like to focus?

 A: Making every page easy/pretty. I would love my job to be reconfiguring the front ends to be more user friendly and modern looking.

 B. Developing an online user network with profiles, skill points and ways to connect power users in a part of the system with other users who need help in that same part of the system through skill point allocation and a connectivity search engine to get the right users in contact.

 C. I would consider shadowing Wayne and learning the server side of things. I don’t know if I would like it or be good at it, but I am interested in learning about it and understand there is a need there. (There is probably someone far better for this job, but I do have interest in learning from Wayne.)

What are your thoughts on look and feel for the application interface?

 A: It is a one of the loudest criticisms I often hear of our software is that it is too busy and hard to read and navigate. “Too many Clicks.”

Let’s talk pricing. What are your thoughts?

 A: I think we are a mature enough company to raise our prices overall, however, I think that expanding the tiered system to include levels of service/access to dev team/consultants/etc. with an overall goal of being inclusive to all sized businesses while still charging the more needy companies enough to be appropriate for their additional requirements.

What about marketing and sales?

 A: Yes?

What about code, databases, automation, testing, deployment, and other techy stuff? Back-ups, redundancy, mirrors, servers, clusters, etc.

 A: I would love to have a database table dedicated to cultivation and harvest and one for packaging that brings in all of the relevant information into one place.

 B. I would love to have a corporate standard unit and integration testing model with dedicated help and standardized system and practices.

 C. I would like to continue and expand the bit bucket pull request code review model.

 D. We should consider refactoring legacy code using more modern coding languages. (Lightly and sometime in the future.)

What are your thoughts on education, training, and add-on services?

 A: I would love to learn systems for unit/integration testing.

 B. Continue Alans Code classes. (during a time of day I can attend. Lol!)

 C. Have classes dedicated to learning all the ways that the users interact the different parts of the system.

What about custom and API sockets and integrating with other external products?

 A: I don’t like trusting companies like Metrc to continue to support the work we do to integrate with them, however, I don’t know how much of a choice we have.

If we do a trust what are the other rules or parameters we want to set up? Is it just time, is it time and money, is it investment, what is necessary if you want to buy in? How do we determine who is active and not active? How many virtual family members do we want? Etc.

 A: Obviously, I hope to be included in the trust when it is implemented. Otherwise, I am not focused so much on money. I will leave this question to those who have put more thought into it.

Where do you see us (as a company) in 1 year, 5 years, 10 years?

 A: 1 year:

1. Unit/integration testing and live user testing in place before every update to the system.
2. All of the major user complaints/bugs in our current services are fixed so Clients are happy.
3. Strong progress on Pretty/Easy implementation.

 B. 5 years:

1. Full social connectivity platform in place and running for some time. Users in need being connected with Users with skills and a fully implemented online payment system where we can collect a small percentage of the payment to the super user for the convenience of identifying them and connecting them to clients.
2. Trust and profit sharing in place. $10,000+ per month to all in the Family.
3. The organization being robust and organized enough so that Steve, Brandon, Wayne and people like them can escape the rat race a little and be able to rely on the other folk to handle more of the burden
4. Expansion of markets and the team so that the overall revenue is higher.

 C. 10 years:

1. We are all hanging out on a beach somewhere while Adilas runs itself. lol

Where do you want to be in 1 year, 5 years, 10 years?

 A: 1 year:

1. I would like to have the cultivation/harvest/packaging stuff working perfectly so that I can be working on Easy/Pretty and the social connectivity platform instead.
2. I would like to have unit tests and integration tests running in the background whenever I am coding so that I know what I am doing is going to work the way I expect.
3. I would like to have a more thorough understanding of the backend/cfc functions so I can code more efficiently.
4. I would like to have enough money to put a down payment on a house.

 B: 5 years:

1. Have the Adilas looking beautiful and everything be easy to use.
2. Have the Social Connectivity platform up and running and bringing in income and easy to maintain.

Maybe have developers that I can help out and do code review for like Alan does. (I will have to learn a lot in order to pull that off. Lol.)

Other ideas and thoughts?

 A: I really love working for Adilas and treasure the work environment on people that I work with. All of the suggestions above are just that and I trust that collectively we will come to the best solutions.