Meeting with Alan, Bryan, Aspen (she was sick today), and I – overarching general plan – day 2

6/20/23

* Ideas from Tommy (Alan’s wife).
  + Free/demo  
    Economy/basic- bare bones  
    Standard/executive - subs  
    Corporate/elite/pro/premium- reports  
    Enterprise/unlimited- cross corp with reports  
      
    Tie usage to feature level… not charge extra/limit throughput  
      
    If they are over for 2 or 3 months automatically bump them up to the next level
* Throttle them and encourage them to go up to the next level
* The free/demo level would have some hard stops… the other levels would encourage them to be bumped up.
* Ski and snowboard levels – true beginner or never ever, beginner, intermediate, advanced, freestyle
* Fun star wars levels - padawan, apprentice, jedi knight, jedi master, sith lord
* Maybe deal with React – JS framework – API backend – frontend components – Have Alan play with some mini obstacle course stuff – R&D
* Creating one-pagers – no action pages just auto save or one-pagers
* We really want our backend to be more like a REST API backend. Currently we do a lot of traditional server-side scripting. The newer way has a separate backend and separate frontend – both levels are object oriented.
* Using an open API socket platform allows for other developers. Like the FlyHi guys – they did a great job building out a full API socket based interface that was super cool – full ecomm, delivery, etc.
* Whatever we build internally, we could expose it externally as well. Do the whole application on the API socket level. Great documentation and we use it and thus maintain it along the way. We have a current API socket system but we don’t really use it as much. We mostly use the API sockets internally through different calls.
* Being able to use functions to build frontend components
* We are talking about rebuilding both the backend and the frontend. Prepping things so that we could use restful API’s.
* Light talks about using frontend tools to transfer the load from our servers to their local computers.
* Bryan asked about a possible converter or conversion kit going from old to new (versions and connections).
* Alan – was talking about being able to fake the backend (mock it) to allow the frontend to do their development. We could also go the opposite way where the backend fakes or mocks what the frontend would do.
* CRUD – create, retrieve or review, update, delete – basically, add/update/show it – database activity
* Talking about levels – at the basic level you need or have access to such and such. We were having fun talking about huts, apartments, standalone houses, and islands. Just being funny.
* Don’t limit the main players… across the board. If they go out wide, you may have to force the upgrade.
* Dollar amounts also play in… what else? Size of business, storage over time, traffic over time, functionality (depth and what you want – bells and wistles), exports, tech support levels, training?
  + We may have certain levels that auto bump you up.
* We looked at the P&L for adilas. We pay for servers, security, storage, and people (coders, developers, bug fixes, tech support, admin, etc.). We also looked at what makes us money (monthly services, storage, special features, custom code, training, project management, etc.)
  + Tommy said – Focus the monthly amount that you charge on what it really goes for or paid out to.
  + Tommy – don’t go too granular… just base it on the tier – not nickel and dime them, just keep it simple and assign them a tier
  + Set thresholds and somewhat force the upgrade so that they can’t use abuse the system
  + Hopefully the lower tier users will want to upgrade to get the better functionality.
  + Let them play with the new functionality without buying it… could be in demo mode or we could give it to them for a few months.
  + Help get our users into the newest functionality.
  + If we have everything in the database – to build out the invoice, we could flag things for free and even roll over other functions and allow them to be demoed for certain periods of time.
* We don’t want people to be overpaying – we want them to feel like they are getting a great deal and a great product.
* This is a real business, we have to charge for what we do. Having confidence in our products.
* Figuring things out by writing a bunch of user stories. So and so wants to do this or is this and needs access to this, how does that work or happen.
* Bryan – Talking about the existing news and updates – we need a way to flag corps for the mmj stuff. We really need this to be at the top level for news and updates. There exists a few settings right now but it is not very obvious. We want to turn things off that we don’t need.
* The user stories were really helping us see the flow and needs and levels.
* Set tiers based on the number of users, product counts, and invoice counts and values. Another big one is how many locations.
* Alan and I were talking about a pricing matrix – playing with levels – wagon, horse and buggy, motorcycle, vehicle, semi truck, sports car, space ship, whatever. That reminded me of the old adilas formula – choose your flavor, pick your tools, skin your engine, build your world.
* On pricing, how quick do we bump them up and/or prompt them. Maybe have a matrix that shows where they are at. Maybe even showing them where they are at over time. Show trends and help them see. Allow for things to go up and down based off of real usage, storage, traffic, etc.
* Once we have some of these docs and matrix pages – send them around and get some good feedback.