See below for other dates as well.

New notes are in yellow – 2/8/16 to 2/11/16

Brainstorming meeting between Brandon and Steve on 10/14/14

* Sub inventory & cost controls (packaging) – sub of time – another function that we can assign to whatever. Virtual adoption process – native for time but available for any other system player.
* Play in bulk or play in individuals – let the users decide
	+ See image
	+ <https://www.adilascontent.biz/top_secret/images/corp22_448057E9CD/pdf/doc_64_06A7694467.gif>
	+ Be able to populate RFID tag or barcode numbers within a range
* Limits (quantity limits) – yes limits or no limits (unlimited)
	+ We still need this – for instance labor – different values for each person
* Attributes – the ability to have additional options, fields, settings, etc.
* Pools – pre-defined rules to help with conversions
	+ 1/8 oz = x g
	+ ¼ oz = x g
	+ Name, price structure, all the conversion rules.
	+ Maybe allow things to be connected to preset rules on the fly… Don’t require them to directly connect things. Allow them to assign on the fly.
* Units of measurement for the package… unchangeable
* Sub inventory items with special values…. (for example)
	+ Labor – mommy object
		- Sub labor – child options
* Price – variable prices and variable quantity to price levels
	+ Some of the special price matrixes may be sub inventory based off of rules and such
	+ Similar to the concept of smart groups – rules and assignments
	+ Help people sell stuff – rules and virtual conversions
* Barcode – We can use a unique number and it will pop up the package details.
	+ We also want to be able to scan or search a parent barcode and pull up all possible subs or child packages. See this file
	+ <https://www.adilascontent.biz/top_secret/images/corp22_448057E9CD/pdf/doc_15967_14173F002D.txt>
	+ Maybe allow for auto incrementing
* Package of packages (mini price and quantity matrix for or per the package) –recipe/builds and templates (processes) from other rules or matrix… think one-to-many – mini manufacturing
* Currently we have a 1 to 1 on price and unit of measurement – we need to extend that…
* Controlled groups – each package is individual
* Pre-packaging
* Scales and integrating with scales – other peripherals (outside pieces)
	+ Barcode thing (parent/child) look-up is needed.
* Pricing matrix per item… without smart group buttons. This could be on a group level or on an individual item level. (rules and assignments or pool subs or recipe subs)
	+ Rules need to be assigned to categories… that could be part categories, sub categories or customer types or both (any – space – organization – or layers)
	+ Be able to add group quantity functionality – similar to the new treat as group option for smart groups.
* Think subs and subs of subs – attributes (individual - micro level) – think of my cart favorite button groups (subs of subs). Maybe flex grid or in-line extensions.
* Be able to package, re-package, and re-package again. Unlimited number of layers or wrappers (how are we applying space)
	+ See new brainstorming in the adilas shop – Time Id’s 624 and 658 – about tracking special subs and details (categories, phases, locations, flags, and tags)
* By default a perfect 1 to 1 on price, quantity, cost, description, unit of measure.
	+ If they want, they could connect to a pre-defined matrix
		- Prices, start/end qty’s, descriptions, and unit of measure.
			* Sub packaging to get to that level.
* On pricing we need price per, and total prices… just like smart group buttons
* Borrow the logic of the smart groups without forcing things to the button level.
* Attributes – Allow them! – empower the users! – Both ways in and out (adding fields and subtracting fields).
	+ Colors, descriptions, sizes, weights,
	+ Sub matrix of other options…
	+ Flex grid at any level… think any attribute – name, use, setup drop downs, rules, defaults, sorts, etc.
	+ We ended up hardcoding 20 text fields, 10 numbers fields, and 10 date fields.
	+ We may need to increase the hardcoded values and/or go to a full one-to-many relationship. We are currently adding 40+ fields per transaction.
	+ We do have some people maxing out the current preset.
	+ Known problem… if we use JSON… we can’t really pin point which of the 40+ fields are the same thing. And we have to redefine the attributes for every category. Ex: size and color – that could be T1 or one category and then T7 for another one. It may help if we have a database base and then use those same attribute pieces over and over again.
	+ Be able to print labels with the attributes. Easier mapping – not in a mixed bundle.
* Packages are very defined… packages may not be as big, but think of a lot full of cars and trucks. Each one is very unique and we need to record that data. The uniqueness, good or bad, may also affect the cost and price (money or the Y scale)
* Part categories and sub categories of categories – stack as needed – we really need this.
* Real In-Line Extensions – we need them – see notes from 10/2/14. Flex grid on the database level.
* Go as deep as we need
* Think of attributes on a per item basis… shoes, balls, t-shirts, sizes, colors, etc. The categories need to be main… The attributes need to be on the sub category or sub level. They ended up being attached to the part category.
* Pattern after the makes and models section (any attribute and make it searchable). Do this on a group level or an individual level.
* The goal is bulk or general to specific or individual. Use the stock/units as a model… They are serialized units… tons of micro details that may be assigned and applied per item or per package…
* Searching space… pretty vast and broad – space could be sub locations, sub phases, sub levels, sub categories, sub special flags, tags, hooks – digital storytelling. Be able to search and get alerts, notifies, updates.
* All of the pieces play in this same realm – all 12 main players, sub line items, payments, sub tables, attributes, and any other sub functionality. Make it even more of a system.

10/21/14

* Go to the very beginning…. Where does it start? We even came back to the planning mode on 2/9/16.
* On multi RFID & Serial Numbers…
	+ Stardard Piece: Ex: XYZ-7777
	+ Auto Increment from N to M. Ex: 400 – 550
	+ Or I need to add X number of these with mini tweaks
	+ Be able to scan an RFID tag number to get to the subs
	+ Be able to scan parent barcodes and get to child subs
	+ If we use time, we could use the RFID tag number as a search option. It might also be good to add new quick search options for subs.
* Stock/Units – tons of 1 to many :: also attributes to parts – tons of 1 to many relationships
* Auto packages – LIFO and FIFO – What is the order of selling? Think of a vending machine… next, next, next…
	+ They get to pick by making a package active/show or active/disabled or inactive/closed.
	+ What if we let them define their own rules and when to open packages and such.
* Settings and overrides – Possible options and how do they play? Automating some of the opening and closing of the packages. This may come later on.
* Possible grouped invoice… In the background we play in detail, in the foreground or to the customer we play in bulk – according to price and description.
	+ Toosie Rolls at $0.05
	+ Toosie Rolls at $0.07 (price is different)
	+ Toosie Rolls X at $0.05 (description is different)
	+ On each child we need to dictate some rules for selling – pricing matrix – tiered pricing at a sub or child level.
	+ Pricing matrixes could use a template and then be used for multiple things and pieces – rules and assignments without having to recreate the matrix. Think about copying rules and assignments and/or assigning different things to different templates. Basically, what pricing template do we belong to?
* Track in detail but show in bulk when possible.

10/22/14

* Building recipes… Build the recipes based on a quantity of 1 – not 12 (what if 10, 11, 12, or 13). On a quantity of 1, it will work every time.
* Build and Sell, Build and Hold, ….. future of Build and Build (stock/unit or a 1 by 1).
	+ What if Build and Build was a set of preset attributes, processes, matrix, attributes, etc.
	+ Think a preset set of rules and functions. You could then use the build and build recipes over and over again.
	+ This may be tied to pools or presets (subs of elements of time)
	+ Remember the costing as well. This is part of building and manufacturing.
* Raw, to work in progress, to finish goods. Run things through settings, processes, or preset lists of things that normally happen. It could also include options. Think of a recipe but then using it on the fly. Adding the attributes on the fly but it all gets connected to a basic recipe. Or we run the recipe (as is with options) but then they can customize the process as needed.
	+ This may also need things like sub locations, sub phases, and other sub tracking options.
* Buttons, checkboxes, groups – Think about building a stock/unit description (lengths, widths, weights, colors, other attributes). Building over and over with standard features.
* 3D level – x=time, y=money, z=space
* On barcodes maybe tie out to categories or allow a switch to determine the action – add to cart, show details, prompt for weight, show matrix, ext. Maybe even a quick screen with other options.
	+ See notes on barcode searches listed above.
* Speed is the main issue.
	+ Tracking the details and/or story is also a huge part of the puzzle.
* Sales and promotions, coupons, marketing campaigns, discount levels, etc.
	+ Each of these things is a loaded statement… and/or a different mini project.
* Scanned like pieces – do quantity discounts based on categories vs. just a simple 1 to 1 sale. Think of smart group buttons but allowing the pricing to be applied to a group instead of individual assignments. Currently, we can assign any number of pieces to a group but they are treated as individual pieces not groups. It may need to go to the next level.
	+ We did take to the next level – treat as a group – only to find it needs to go to a smart cart logic level – custom rules - it still needs to go deeper.
* Sub inventory pieces (tied to customers or promised for certain jobs) – Basically putting some of these products on reserve and then special pricing for certain customers. Assigning products out before they are really used… kind of who gets what and from where?
	+ Promises, preorders, job costing, contract pricing
* Attributes may need to be on an invoice. Let’s set up things so that it can play as such. Think templates and sub functions. What do you want? Play on the full custom level. Build your own….
* Custom labels – We may need to add ingredients, stats, attributes, conversions, package details, etc. As a note, labels don’t have to be a sticker… it could be any other documentation, pages, notes, copy, verbage, etc.
* Recipes…. That could be the process, the pre-set values, the defaults, the verbage, the contents, any other documentation pieces. Think any other data stickers or otherwise… Data sheets, photos, instructions, warranty, disclosures, etc.
	+ Maybe connect recipes to media/content and files. And other important information.
* Breaking traditional accounting again! Professional Day Dreamers! We also try to put it in to action.
* Business Zipper – 3D World Building – A complete system – Building the bridges between the different players, groups, and pieces. It is a system.
	+ Data assembly line and dynamic data engines

10/23/14

* On in-line extensions… We need to be able to set which field is required, what order to show, show/hide options, defaults, add or subtract the new ones.
* API socket level … play at the wall – wall outlet or surge proctor level. Application plug-in
* The rising tide raises all boats – Maybe present some community funded projects. Show people the value of where we are headed.
* 3rd Party solutions – “Let them play at the wall” – Wayne Moore
* Virtual cash box that is automated.
* Cost of goods to inventory (COGS) – we will have this piece.
* Next logical step – This is what our goal is… What is the next logical step?
* We are selling a partially done product… That has been a huge pain but super worth it… We have a producing idea farm – right now! Our clients are helping us drive the ship! – If we have a producing idea farm – we need to harvest some of these things.
* We try to listen to what our customers are really asking – within limits of budgets and time – what can we do to help the ones who love what we do? – we may need to pick and choose what clients we service. – what will benefit everybody the most?
* We are the train tracks – we’ll let others build out the other pieces as they see fit and needs.
* Quote from Danny Shuford – Adilas is a business solution for solution minded people.
* Ideas from Ryan Fox – Adilas Community – Post things – Training – Teach from the inside out – This is a huge resource that we could tap into – Tap into the power users – Help them setup their own companies – We are sitting on the answer to tech support and customer service – basically make a call to the power users and see if they want to help make some side money – allow our people to make some extra money – allow them to set their own schedules or times when they are not working for someone else
* What if we had a merchant account that could be used for the adilas community. Help push things around.
* Be able to send requests to users for help and tech support – training – setup – audience that is watching what is going on – maybe use Calvin’s gmext texting to the power users.
* See elements of time – in the adilas shop - # 667 – concepts of sharing resources and tapping into a community type platform.
* Referrals and testimonials – who knows it, who loves it, who can help us spread it out.
* Idea on a flat tech support/training fee - $20/hour for our community to help with training, setup, and such. We (adilas) then would pass the monies around. We would eat the credit card fees as our part of the process. We could spread the monies around and help track things.
* The way we make our money is by more and more clients using our system. There will be tons of ways to make other monies once things get pushed and get going.
* We really need to get the job costing and elements of time thing done to a higher level… That would be awesome.
* What if we change from “as is” to “community funded project” or “community driven project” – What do you want? And what tools do you need? Here is what we have, if you like it great. If you want something else, let us know. Let’s drive this thing together.
* Another internal asset is the data that we are storing and connecting together. That is huge!
* Whoever wants to play… then let’s play! That is who we want.
* Get the system to help the people know what to do next and how to fix potential problems that come up. Easy wizard steps, new interfaces, training (inside out), new setting and features.
* Corp setting, page settings, location settings, user settings, manager settings, etc. To what level? Business world building!
* Business to Business (B2B) transactions. Application flex grid options. EDI electronic data interchanges.
* Keep plugging, keep plugging! We are going in the right direction! The windows keep opening. This project is way bigger than we are… ☺
* Our business model is: small to medium sized business, self-server, software as a service, open API model, etc.

6/17/15

* We need to add new database columns to the time template table.
	+ use\_sub\_inventory tinyint(1) 0
	+ sub\_inventory\_tracking\_name varchar(50) Null
		- Default to RFID Tag, Lot #, Batch #, Package #, Group #, Heat Code #, Mixture #, Canister #, Case #, Pallet #, Bin #, or whatever… (variable or a setting)
		- We ended up calling it sub inventory, package, and parent/child inventory. This could change but this is where things are at currently.
	+ sub\_inventory\_barcode\_start varchar(5) Null (Internal barcode generator has a max of 15 characters that I can encode. First character must be an alpha or special.)
		- A\_, ~, pkg-, p\_, etc. Sample: A\_10, A\_111119, ~10, ~111119, A\_5555555555555, X5555555, p8111111, a15061755555555
		- We ended up going with the ~ for now.
	+ sub\_inventory\_barcode\_top varchar(100) Null - Mapping for top line
	+ sub\_inventory\_barcode\_bottom\_left varchar(50) Null – Mapping for bottom left
	+ sub\_inventory\_barcode\_bottom\_right varchar(50) Null – Mapping for bottom right
	+ sub\_inventory\_barcode\_description varchar(255) Null – Mapping for description
	+ This whole section may need some lovin to get all of the pieces that we need. These were just some starting ideas.

7/27/15 more notes from phone call and GoToMeeting session between Garrett, Steve, and Brandon

* We may need a new setting and/or permission. Not everybody needs this feature and it may be nice to charge extra for it… This is still possible – we may want people just to use it as a standard feature. This will become the norm if we get it done good enough. We are thinking this needs to be a core component.
* On the Excel spread sheet, we added 6 new possible database fields to the sub inventory table. They were a sub\_alias\_number (what do they want to call the sub id – also used for a barcode value), use\_expiration\_date (tiny switch to say use as a real date or not), sub\_size, sub\_size\_2, sub\_color, sub\_color\_2.
* Garrett is going to look into woo commerce and see how they do some of their product matrix stuff.

2/9/16

* Prioritize the needs based on budgets, needs, and wants
* Allow a pioneering team help cut the path and then circle back around and do another round or two. On purpose – do trystorming
* Sub part categories – may be a huge first step. Let’s do this as part of the next session.
* Community effort – keep building and dreaming – help businesses succeed – kinda like a bee hive where we attract other worker bees to help in the process.
* Education and maintenance – those may be more important than new features – maybe tap into the internal user pool or from the inside out. See note above on this. Great idea!
* Find the sweet spot – mix between monies, workers, and dreams. Having your whole team there…
* We are into business progression – what is the next step? We’ll even help show them the map or show them what we are doing – create that adilas community
* Having sub inventory work properly is a huge key of running a business and tracking your inventory – keep helping people.
* Workshops to explain what we bring to the table and what we offer – small advertising and getting things out there. I would recommend that we focus on the core business functions and how we can help our clients.

2/11/16

* As a note, there is a huge section up under 10/23/14 section that was added on 2/11/16. The new stuff deals with some new ideas on training, tech support, and building out the adilas community. See notes above for more info.

5/14/16

* I would really like to start over on sub inventory… include all of the above pieces and rework the entire things from the ground up. We’ve had struggle after struggle and I don’t feel comfortable building on what we have. We have paid for a path through the forest. That is what we have at this point. I would like to make it a highway.
* See emails between Steve and Michael Webber (McCorvey’s Bowling). Some good information. Another good resource is Karen Nab (adilas consultant).
* If doing this project, we really need to start at the very beginning. This could be clear down to the sub part category level. We have also already had huge needs for this outside of the secured environment in the ecommerce and API land. Make it a real highway and a platform to build on.
* I may need to learn more about JSON, JQuery, and such. Don’t worry, get in there and figure things out. It is ok to build side pages or mini test pages to get the concepts.
* Custom barcodes seems to be a big issue. Check in to that a bit more. We’ve also had requests to be able to use the parent barcodes and be able to tie-in and show the subs.
* Make the sizes and colors, a simple and easy to use set of attributes. Fully review, plan things out, and start back at the beginning.
* If the route needs to change, that is ok. Remember, you currently have a small path through the forest. Make it into a road or highway. Think broad – Almost every industry could use this technology.
* There are some known issues with the current code set. Make sure and fix the units of measure, the dates, the RFID tags, the cost, etc. The cost is a big part of this whole puzzle. Make sure and include that.
* Am I coding this thing or am I trying to coordinate it? That is a big question. Who is going to code it? Who is going to plan it out? Who is going to sign-off on it?
* What is more important – Splitting up databases, world building, or fixing sub inventory? Not totally sure at this point.
* Be your own style!

8/1/16

GoToMeeting with Steve, Danny, Russell, Nick, Alan, and Brandon

- original name - sub inventory and cost controls

- parent/child relationships

- packages, batches, heat codes, chain of custody

- RFID tag tracking

- barcode issues

- extra details

- sub rules - show/hide, flow, automation, look-n-feel

- sub attributes (one to many - unlimited) with data types (text, numeric, date/time)

- labels - custom positioning, formatting, attributes

- tie-in with elements of time

- quantity rules

- known issues

- data migrations

- standarization of code

- core vs. extra fluff

- function vs. fashion - usabilty is huge

- templates, sub part categories

- conversions and visual aliases

- new code from Calvin and working with older files

- this is our next level - we need to make it good

- verbage changes

 - attribute type (old) - data type (new) - these are things like text fields, drop-down list, numerics, date/times,

etc.

- price increases for this feature

- template price matrixes, group pricing, tiered pricing, rules and assignments - like smart group buttons without making

buttons

- search options

- operations leading into good accounting - cause and effect relationships

- client feedback - what an asset

- display issues - horizontal vs. vertical, older forms and submittings or newer AJAX and JQuery (javascript stuff)

 - CSS options and making things noralized

- keep it simple - sales side vs. the backend technical stuff - be careful how much we rock the boat - we have a number of

people already using things.

- user experience - training and making it easy to use - short cuts, copy/paste, pre-sets, borrowing between templates

- nail down the process, set things up, bring it in, be able to see and sell it, show report, go clear to accounting

- maybe think more about possible templates - help setup things for speed and bulk

- inbound templates, selling templates

- locations, sub locations, sub phases, sub flags and tags

- peole need consistency, speed, style, and power

- templates, functions, settings, permissions, display models

- focus on functionality and usability

- lower the number of clicks, make presets and templates

- visual flow, step by step, full process, map it out, get organized, route finding (basecamps), pitches and levels

- fix the existing trail - keep the traffic going

- Expand "who" wants to help contribute to the plan - Kelly, Karen, Danny, Steve, Russell, Molly, Chris Johnnie, etc.

- be able to control the numbers a little bit better. This could be elements of time, or a whole new sub numbering system.

 - certain numbers for lots

 - certain numbers for batches

 - certain numbers for packages

Our own project management piece... (ideas from Steve)

1. Bringing inventory into the system

 - pages

 - templates

 - thought processes

 - steps

 - labeling

 - ordering

2. Holding that inventory

 - maintaining

 - adjusting as things happen - flex

 - accounting

 - labeling

3. Selling the inventory

 - internal

 - POS

 - ecommerce

 - buttons, barcodes, pricing matrixes, rules, & assignments

 - labeling

4. Reporting and results

 - where are things at?

 - what do I have?

5. What else??? Keep it open and available

- on the existing trail - addEditSubInventory - method on PO side and the on the invoice side.

5/8/17 – new brainstorming session.

* Rules for child inventory – show, hide, open, close, etc.
* Sub part categories
* Conversion stuff needs to play into sub categories
* Packaging and packaging of packaging – it could get pretty deep.
* The pricing engine – sub categories may be the basis of the pricing tier
* Conversions need to play into the pricing matrix.
* Elements of time and sub phases and sub locations
* We are seeing that we may need to build out the conversion process before circling back around to wrap up parent/child/sub relationships. Think of rough cutting a trail through the forest. See where we could go and what we could get to. Look around and let people start playing, even if it is a little rough. Another analogy is keep framing out the house before you start picking colors and doing the finish work on the kitchen.
* We need processes of bringing things in, holding it, and selling it.
* Often we talk conversions… What about reconversions (going backwards or forwards)?
* We are having more and more pressure out in the ecommerce land. People want sub categories, ability to show sub categories, show sub inventory, show conversions, show pricing matrix, etc. It is all interconnected – very interesting.
* Turn the main view parts (view parent) page into a virtual hub for the item. We need to show parent/child, sub inventory, conversions, smart groups, my cart favorite buttons, etc.
* Since the start of this project, we have learned tons. That is super awesome. Keep gathering things, ideas, and concepts. Apply things as we go.
* What if we start hanging our pricing model (pricing engine) based on sub categories. We don’t even have those yet, but they are somewhat of a bridge of sorts.
* On conversions, we need to be able to build things up and break things back down. Basically, go in and out on the conversions.
* We have processes that need to be broken down prior and things that need to be broken down just in time. How does that tie into reporting? How many do I need or how many could I make?
* Data and trending? What was sold? What was made? What is popular? When and how fast were they sold? What was the price point? What outside events were happening? What were the averages? Reporting past, present, and future projections.
* On conversions, what is made, what is on hold, what is sold?
* On conversion and what is on hold, how quick do we clear that? Automation process? Manual updates? Etc.
* What about limits and pricing models for things out in ecommerce land? Do we allow this? What about that? What are the rules? Where do we store those rules? What about discounting? Does this category or sub category allow discounting? If yes, what is it?
* Settings on show/hide, use/don’t use, and other rules (settings).
* We need great operations that go clear out to the accounting level. Tell the story. If you don’t catch and tell the story, the numbers don’t mean anything. Digital story telling.
* All conversions will come from sub inventory unless the parent item is set to unlimited (labor and services). That is the only way we allow conversions to be tied directly to a parent item.

5/16/17

* Steve and I were talking about specific attribute names. Sometimes we need to lock things down and require certain things. As an idea, we would like to start setting up accounts and have basic presets we could load as needed. That would help us standardize things.
* Chart of accounts – this is an accounting term – we may need a similar thing for attributes. Don’t worry about the numbers so much… maybe think more about the word “chart” or “virtual mapping”. That could really help.
* We don’t want to tie things to the names… people like to change those all the time. We almost need a reference number, a special flag or special tag field, or some other special mapping field.
* We are seeing a need for histories for subs, sub inventory, and even sub templates and attributes. If people change things, we need to know what was changes and who changed it.
* Maybe take things out to the level of the db\_field\_settings stuff. This is a table that allows for actual names (behind the scenes), aliases, data types, required (yes/no), show/hide, min/max, sorts, special instructions, etc. The current attributes have a bunch of these of these same values. Maybe mix and blend some of these tables (db\_field\_settings and sub\_inventory\_attributes).
* Who is going to make the chart and/or the original mapping? This will end up being needed for later developers to pull things out of the API. We need some kind of developer hook and/or place we can virtually store a chart of account number for the sub attributes. Almost a pre-mapping.
* The other thing we could do is a mapping… x in system a = y in system b. Maybe the application flex grid type level.