Bootstrap migration

1/24/22

Chuck, John, Brandon

* Maybe going from Bootstrap 3 to Bootstrap 5 (?)
* If we do this, we need to replace all jQuery with vanilla JavaScript
  + Date pickers, predictive select (chosen), tons of other things
  + Chuck said we could still use jQuery, we just need to pull it in on a page level
* Other jQuery pieces – such as the color pickers
* SummerNote – html editor
* What about datatables?
  + Currently they are using a plugin to do print, export, etc.
* Patch the boat or build a new boat?
* Prototypes and samples – aka templates and patterns
* Component review and style those out
  + Buttons, submit buttons, inputs, radio buttons, checkboxes, text areas, toggles, file upload, etc.
* Other developers – we need all of the developers to use and follow the docs. A fixed or standard package or style guide.
* Better code review – they have to follow the set standards
  + On the sprints, it seemed to be easier to manage it and really stick to an accountability
  + Does someone, who knows the standards, review and sign-off on the code.
  + Follow the styles and style guides
  + We really need a dedicated or second set of eyes that looks at the code that goes through
  + Providing a physical or virtual checklist for the devs
  + We need to slowdown a bit… it feels like we are always pushing things to the limit or to the max.
    - What about a backend review and a frontend review?
* We have a need for full stack developers – backend, frontend, and everything else in between
* We have a lot of single heroes and lone wolves – wild west vs more standard and strict
  + Create and follow style rules, code rules, project rules, etc.
* We are seeing a need for a full cultural shift
* There are lots of ways to reach the end… what is the best way for adilas?
* There is a need for internal education
* Possible entry tests or a series of tests – different cert levels
* We need more of…
  + Frontend – requirements and specs – what is wanted and needed – project management stuff
  + Backend – code review and deployment stuff
  + We could use both John and Chuck to help with code review
* We need less of…
  + Timeline promises – we need to run things through real processes vs forcing it all the time
* Extra charge for rush jobs.
* Make a plan for accountability and following the standards
  + Let them know
  + Get them trained up on the new processes
  + Hold them accountable
* From Brandon – Let John and Chuck come up with some rules and standards. Then run it back past Brandon and Steve.
* Enforcing the standards
* Maybe making a pre-sign-off checklist or something like that. Post it in the docs and use it for sign-off and code review.
* Looking into the new non-compete and non-disclosure doc – see other doc – we’d love some feedback.
* Lifespans on themes
* Stopgap solutions – bloating the system – future maintenance needs
* If we keep going the way we are going, the current ship that we are sailing is going to sink. Looking to the future – what is going to happen in 6 months? 1 year? 3 years? 5, 10, etc.
* Pretend that there is a rough budget of $80,000/month. We have to figure out people, servers, maintenance, future, repayment, etc. What is our plan?
* We need to start building or heading towards a newer (tech) plan – there is a lifespan and shelf life on code and tech.
  + There will be a need to rebuild the system every 10 years (at least). It may be only 3-5 years. We have the code, tech, software side, and server side of the puzzle.
* Savings and investments – looking and planning for the future
* We need to pass on the costs to our clients. We have to increase our prices.
* What if we took some percentage or dollar figure and start putting it into the build a new ship fund. Manage those funds and future growth projects.
* Get Steve involved in some of these decisions.
* Getting Chuck and John in on the code sign-off should help with consistency and relieve some load from Brandon. Enforce the standards.