Blueprint for Building a World-class UI

The Role of the user Interface

The UI is more than just the visual components. It's the only way your users will interact with your product and its functionality. It is fundamental to the experience that your clients will associate with your company.

Start at the end

Start by defining the outcomes. Work backwards from how you want the various stakeholders to describe the product after it's been in production for several years.

Solving for now, solving for then

Thinking long-term is vital as it could be years before you get to change course again. In addition to defining a strategy that will win in the future there's also likely to be a more immediate need, and that is to close the gap with the competition. A good roadmap will cater for both.

Learn from the past to solve the future

With the benefit of hindsight, UI's built over the last couple of decades failed for a few specific reasons and it's important to take onboard those lessons; i) Existing UI's were visualizations or projections of the system's functionality whereas they should have been a reflection of the user's workflow, ii) UI's didn't cater for the future they solved for a point in time and so fast became unfit for purpose and iii) some skills required to build a great UI differ from the backend skillset.

What do you want your clients to say?

In 5 years time, if you were to interview your clients about their experiences using the UI, what answers would you want to get ? Here are some examples:

- Allows me to spend more time trading and less operating.
- Adapts to my way of working not the other way around.
- A competitive advantage not a hinderance that I work around.
- Makes me better at my job.
- I can rely on it to work when and how I need it
- I can get the changes I need really easily
- Integrates well with my data and systems
- Leverages modern technologies and techniques

Cost efficient and fast to change

Two sides of the same coin. The ultimate test for any new product will be the test of time which will depend on how easy it is to change and adapt but also how much it adds or takes away from overall profitability.

Having a solution that's easy to learn and be productive on will help in adding new developers and make less experienced developers productive. Similarly, testability, deployability and supportability will be key given they can count for up to 66% of software costs.

Keep up with technology advances

Although the web stack seems quite stable new technologies will continue to emerge. Any new product design needs to be as modular as possible to enable simpler swap outs for better tech as it comes along.

Promotes Reuse of visual components

Applies to both of the above but is also worthy of its own section. Polishing visual components can be extremely costly and inconsistent look-and-feel will kill the user experience. A clear reuse strategy is required to keep costs down and agility up. Deciding on the right level of granularity and what is common is vital. Too coarse-grain will limit reusability but too granular will be hard to maintain.

Appealing to advanced and less advanced clients

Most front office users need a system to aggregate data from many sources, to be analyzed and acted upon. Historically this has been a predominantly manual activity and different asset classes are in different stages of migration to more digitized, exception-based ways of working.

A new system needs to cater across the client spectrum, from users who are slow to adopt new approaches to users that want to be on the bleeding edge. It's pretty much a given that over the next 5 to 10 years, cost and regulatory pressures will push automation as far as it can go and what gets built now will need to handle it.

Recognize why your current UI is so far behind anyway ?

A good question to ask is – why do we even need to do this exercise? Ignoring budget restrictions, why didn't the UI evolve along with the backend functionality?

Generally, it's because the UI has always been an afterthought, a lower priority than the backend. Historically UI engineering was not recognized as being technically challenging and typical skillsets did not address the artistic and creative elements.

There can be only one

The need to maintain one user experience across multiple products requires org changes. Conway's Law is a fact of life and although it's possible to maintain more than one consistent experience across multiple teams, it will create more work and risk to do it that way.

Taxonomy and governance

Maintaining a consistent UX will come with an administrative cost. A functional, empowered design council will be required to own UX integrity. One element of this will be deciding whether a requirement for a new visual component really represents some thing new or not. Clear taxonomy and design guidelines are required here to maintain the house view on what's common and what's special

Invest in a UX designer

Big Tech and the Internet have completely changed the landscape and expectations on what a UI can do and today consumer technology users have highly usable, works-of-art in their pockets. Keeping up requires much more than the traditional software engineering skillset and specificialized creative resources need to be added to the team.

Features of a World-class UI

Starting with the basics

Maximum configurability

Any and all visual formatting or style should be accessible by the user, from macro elements like workspace, to micro elements like font type or row height. Users should be able to maintain multiple different configurations and switch easily between them.

End-client viewable

Client facilitation businesses should be able to configure and permission tear-off views on the fly and make them accessible to their end-clients, making interactions more seamless.

One UI with one way of working

For products with a broad set of user personas, irrespective of function, there should be a consistent experience, that's familiar and logical to the user. Like keyboard auto-complete, the UI should be able to complete the users sentences.

MS EXCEL like

Excel is the most successful piece of business software ever. Copying and improving on some of the more powerful parts is a winning strategy. Its ability to enable non-technical users to create powerful tools and workflows on the desktop paved the way for today's low-code products.

For the grid components, imitating EXCEL features, like being able to create powerful derived data columns using formulas and multi-dimensional pivot tables, are a must.

Monetizing data

EXCEL is also great at enabling data monetization but struggles with real-time data. A new UI needs to make sure that any available data, should be accessible and usable anywhere and remain responsive, irrespective of the data volume. Examples are :

- Grids of data pivotable on multiple levels.
- Calculated columns defined as functions of other columns
- New data grids created by joining existing grids
- Complex filters and sorting
- Extensive array of charting and different data visualization tools.
- Unlimited number of rows and columns

User oriented means workflow oriented

A modern system needs to be designed from the user down and the user's workflow involves other apps on the desktop:

- Communication. Two-way integration with chat, email, video, phone should all be possible without having to leave the system container.
- Collaboration. Pretty much all trading activity involves collaboration, with clients, with colleagues. The system should facilitate this as much as possible. Sharing workflow, sharing configuration, sharing context.
- Data integration. Whether its from spreadsheets, system API's or the cloud

Social sharing

Social media will continue to gain momentum, being able to seamlessly consume and post content will be important.

Advanced Features

Serving the End-user that can code

End-user skillsets have seen a steady shift towards technology over the last decade. The rigidity of most desktop trading systems, combined with the need for rapid innovation, have forced able front-office staff to overuse EXCEL and other UDA's (User Defined Applications). The desktop tools of the future will need to provide a similar sandbox but with the right levels of control and governance (source code control, auditability, resilience etc.)

Decision orientated – Going Beyond the grid

The grid or blotter has been the staple data visualization for at least 25 years. The blotter is "data orientated" and it drives a do-it-yourself workflow: here is the data, go find information and then act on your decisions. Modern UI's will enhance productivity by being decision orientated and exception based. The user can define rules, the system can operate autonomously and notify when it doesn't know what to do. Notifications would be highly actionable - "Here are decisions you could make, here is the data that supports the decision along with the ability to act".

Power search – Going beyond the mouse and the menu

Mouse control in general is much less precise than the keyboard and finding functionality by traversing menus is hard, even when you know what you are looking for. All "search" should be possible from an intelligent text-based search function. Whether you are looking for features to use, looking for trades that match certain criteria or setting up an exception-based workflow rule, all this can be driven much more efficiently from search.

Leveraging working patterns with machine learning

Most people's working patterns are repetitive but also unique. User clicks, behaviors, searches should be stored, and machine learning algorithms employed to predict what a user is likely to want to do next and then render useful information and functions on the screen in anticipation.

Finer grain visual components

To enable much of the above the UI needs to be quite dynamic and able to group together different visual components. This means the traditional coarse grain components need to be carefully broken up into more fundamental building blocks that can be joined together in different ways. For instance, The order ticket needs to be 1 configuration of the 20 different visual elements.

Contact us to find out how to #buildfaster.

FRANKFURT

Rahmhofstr. 2-4
60313 Frankfurt
am Main
Germany

NEW YORK

500 7th Ave
New York
USA

LONDON

City Place House
55 Basinghall Street
London
U.K.
EC2V 5DU

Velox Financial Technology was founded 5 years ago by ex. Goldman Sachs Managing Directors. Their mission is to create a new approach to building high performance data-driven software that can help engineers everywhere to #buildfaster.