Inspired by Jordan Peterson’s *12 Rules for Life* I was struck by how his principles of order, responsibility, and personal growth could transcend their original context and apply to various aspects of life. So I started analysis, rule by rule, to see if it would work. Especially interesting was the idea that small, consistent actions lead to significant results, as it mirrors the iterative process in software development, especially Agile. I became curious to see if these rules, which advocate for discipline, clear goals, and a focus on the fundamentals, could also enhance the way we approach coding, project management, and team collaboration. This curiosity led me to explore how these principles could be integrated into the world of software, driving better outcomes and more purposeful work.

When going through rules I adjusted a bit of wording for each rule so it better represents aspects it touches for businesses and software houses. Let me show you how.

1. Stand up straight, when doing business or managing project

In the world of software development, “standing up straight” means taking ownership of your projects and maintaining a posture of confidence and leadership. This involves setting clear expectations, being transparent with clients, and confidently guiding them through the development process. It should be considered on both: by client taking responsibility for his idea and software house – by delivering not only software but VALUE with confidence.
For example, when we assess a project, we don’t just throw out a range of estimates. We take the time to explain the rationale behind our assessment, ensuring that our clients understand the scope and potential challenges. This builds trust and positions us as the confident leaders they can rely on.
For instance, in a recent project Profit Board (https://websailing-new.stagetest.it/pl/case-studies/profitboard/), we helped a client navigate complex technical requirements by standing firm on the necessary steps for a scalable and secure SaaS solution. At Sailing Byte, we believe that strong, clear leadership is essential for successful project delivery.

2. Treat yourself with same respect you give

other

In software development, treating yourself well means valuing your own time, skills, and expertise. As a software house owner and Product Owner, I often see the importance of self-care mirrored in how we handle projects. By treating our team and ourselves with respect, we create an environment where quality work thrives. It is often being said by business owners and yet still underestimated: remember about self-care!
In practice, this means setting realistic deadlines, avoiding burnout, and ensuring that our team has the resources and support they need. In the long run it is most beneficial not only for you but for your clients aswell! For example, in a recent CRM project (https://websailing-new.stagetest.it/pl/case-studies/lourse-international/), we demonstrated this by pacing the work in a way that allowed for creative problem-solving without compromising our well-being or the quality of the final product.

3. Make friends with your clients as you grow with them

In the context of software development, this rule is about building relationships with clients who are aligned with your values and goals. At Sailing Byte, we aim to work with clients who are invested in creating something meaningful, not just quick wins. We do understand that best relations will last long and benefit all sides – if client grows, we can grow aswell and vice versa.
We foster these relationships by being transparent about our processes, like Agile and Scrum, and by providing additional hints what can be improved in their businesses. When both parties are committed to the success of a project, the outcome is always better.
A great example of this is that we are working with a client KDK(https://websailing-new.stagetest.it/pl/case-studies/klinika-dla-kobiet/) who is equally passionate about the user experience as we were. Our collaboration led to a product that exceeded both our expectations and those of the end-users. And our cooperation is still blooming.

4. Compare your business to who you were

yesterday, not to who someone else is today.

This rule is a reminder to focus on continuous improvement rather than getting caught up in comparisons. In software development, this translates to iterating on your processes, refining your skills, and constantly seeking ways to improve your product offerings. And with Agile EBM context it strengthens reliability on evidence and adjusting to current situation against situation that was valid yesterday.
At Sailing Byte, we regularly reflect on our past projects to identify areas where we can improve. Whether it's optimizing our use of Laravel and ReactJS for better performance or refining our Scrum practices, we're always looking for ways to do better than we did yesterday. And we are helping our clients to understand evidence based management, so they can try this aswell.
For example TenantTracks (https://websailing-new.stagetest.it/pl/case-studies/tenanttracks/) project is where we improved our CI/CD pipeline is a perfect example. By analyzing past deployments, we identified bottlenecks and implemented changes that resulted in faster, more reliable updates.

5. Do not let your clients do anything that makes you dislike them

The key here is to set boundaries and guidelines that prevent projects from spiraling out of control. This means educating clients on best practices, helping them set realistic expectations, and ensuring that the project stays aligned with its original goals. Many of our clients at the very beginning had no idea how to run SaaS business, but since we guided them with some aspects they are now thriving. This can also be understood that you should only run business that you actually like!
At Sailing Byte, we use workshops and detailed planning sessions to prevent scope creep and keep projects on track. By establishing clear guidelines from the start, we avoid the frustration that comes from misaligned expectations.
For example, in TenantTracks project (https://websailing-new.stagetest.it/pl/case-studies/tenanttracks/) we helped a client understand the importance of prioritizing features based on user behavior, we avoided unnecessary complications and improved product that met their needs without going off course.

6. Set your business in perfect order before you criticize the world

Before advising clients or making recommendations, it's crucial to ensure that your internal processes are rock solid. At Sailing Byte, we take this to heart by maintaining a strong internal structure—whether it's our adherence to GitFlow, our deployment strategies, or our security protocols. Another example is having standardized backup procedures or CI/CD deployments.
By setting our "house" in perfect order, we are in a stronger position to offer valuable insights to our clients. This also means that when challenges arise, we are well-prepared to handle them.
In a recent project for Axim Design (https://websailing-new.stagetest.it/pl/case-studies/axim-creative/) we tackled a security issue that our well-ordered internal processes allowed us to find and address quickly and effectively, ensuring the client's trust and satisfaction.

7. Pursue actions that provide value

In software development, it's tempting to cut corners or choose quick fixes, but these often lead to bigger problems down the line. At Sailing Byte, we prioritize meaningful, long-term solutions over short-term expedience. This means choosing scalable technologies like Laravel and ReactJS and adhering to best practices in software architecture. This also means that we are sometimes choosy about projects we actually want to make. We want to make the difference, we want to make things that matter.
That also means, we encourage our clients to focus on what will bring long-term value, even if it takes more time or resources upfront. This approach has led to more robust, scalable, and successful products.
For instance, in NFF project (https://websailing-new.stagetest.it/pl/case-studies/new-food-finance-pl/) we opted for a more complex, but ultimately more scalable, database architecture, and client saw significant benefits in the product's performance and flexibility over time.

8. Tell the truth – or, at least, don't lie

This rule is so universal that it does not need any adjustments.
Honesty is foundational to successful client relationships and project outcomes. At

Sailing Byte, we believe in being upfront with our clients about what can and cannot be achieved, potential risks, and the realities of software development. It is also a fundamental rule for proper Agile development – as long as we see each other with client as professionals, it is working well.
This honesty extends to how we communicate throughout the project lifecycle. If a feature is going to take longer than expected or if we encounter an unexpected challenge, we inform the client immediately and discuss how to move forward. This sometimes means a lot of communication by Slack or email or on meetings, but ultimately it is best approach we have found up to date.
In a recent project Animala (https://websailing-new.stagetest.it/pl/case-studies/animala/) where we had to renegotiate a timeline due to unforeseen technical challenges, our commitment to transparency helped maintain the client's trust and ensured the project continued on a strong footing.

9. Assume that the person you are listening to might know something you don't

As a Product Owner, it's important to recognize that clients bring valuable insights about their industry and users. At Sailing Byte, we listen carefully to our clients, valuing their knowledge and experience, and integrating it into the development process.
This principle is especially relevant during our workshops and planning sessions, where we encourage clients to share their vision and expertise. By combining their knowledge with our technical skills, we create solutions that are not only technically sound but also aligned with market needs. This also helps to reflect clients on their business idea and confront them (sometimes first time!) with someone who has experience launching SaaS.
In MyRotat project (https://websailing-new.stagetest.it/pl/case-studies/myrotat/) where we collaborated closely with a client who had deep industry knowledge, the result was a highly targeted SaaS product that quickly gained traction in the market.

10. Be precise in your speech (and emails)

Clarity in communication is crucial in software development. Whether it's defining project requirements, writing user stories, or discussing technical details, precision

prevents misunderstandings and ensures everyone is on the same page.
I do also prefer written communication instead of spoken when it is required to be precise – both because it allows you always to go back to message, and because it allows reader to read few times same words to understand them better. It is not possible with spoken words and spoken words can sometimes be remembered wrongly by other person.
At Sailing Byte, we emphasize precise communication at every stage of the project. This includes our detailed assessments, clear documentation, and regular updates during sprints on Asana. By being precise, we minimize the risk of errors and keep projects on track.
Take for example system with multiple integrations where you need to communicate with multiple parties at the same time – that was the case for CargoSeller project (https://websailing-new.stagetest.it/pl/case-studies/cargoseller/). We implemented a complex integration, our precise communication with the client, with API providers and within our team was key to avoiding costly mistakes and ensuring a smooth development and deployment.

11. Do not bother clients when they are experimenting

Clients and end-users, like children learning to skateboard, need the freedom to explore, experiment, and even make mistakes. At Sailing Byte, we recognize the importance of giving clients the space to innovate and try new things, even if it means taking some risks. It is part of EBM aswell – try to experiment and see what happens!
We fully support this by creating flexible, scalable solutions that allow for experimentation. Whether it’s a new feature or a different approach to marketing, we encourage our clients to take calculated risks, knowing we’re here to support them.
In a WISSP project (https://websailing-new.stagetest.it/pl/case-studies/wissp/) we worked with a startup trying out a novel business model, we provided the technical foundation that allowed them to pivot and iterate quickly without losing momentum.

12. Take step back and pet a cat

In the fast-paced world of software development, it’s important to appreciate the small, often overlooked moments of success and joy. At Sailing Byte, we make it a

point to celebrate milestones, whether it’s the completion of a sprint, a successful launch, or positive client feedback.
Taking time to “pet the cat” reminds us of why we do what we do and helps maintain team morale. It’s these small moments of appreciation that keep us motivated and connected to the bigger picture.
If you reach a milestone, take the time to celebrate with the team and the client, reinforcing our commitment to both quality and enjoyment in our work. Be proud and happy of your work!

Can 12 rules of life be 12 rules of software development?

Most interesting thing in this analysis was that I have already applied those rules to both business and software development processes, which leads me to the conclusion that 12 rules of life are quite compatible with Agile software development. For sure – there are more in 12 rules that are not present in Agile and vice versa (like for example explicitly stated Product Goal). But in my opinion it is quite amazing that you can somewhat overlap those two ideas and add much more value to both personal life and business development.