

# Coach Freelancing Platform Project for Software Engineering Semester 2

Francesco Schenone      Kacper Grzyb      Levente Sohar  
Sebestyen Deak      Ignat Bozhinov

2024-02-24

Syddansk Universitet

Coach Freelancing Platform  
Project for Software Engineering  
Semester 2

Faculty of Engineering  
BSc in Software Engineering  
Project period: 2024.02.01 – 2024.06.07.

Participants:

**Kacper Grzyb** (kagr23@student.sdu.dk)  
**Francesco Schenone** (frsch23@student.sdu.dk)  
**Levente Sohár** (lesoh23@student.sdu.dk)  
**Ignat Bozhinov** (igboz23@student.sdu.dk)  
**Sebestyén Deák** (sedea23@student.sdu.dk)

# Contents

0.1	Introduction . . . . .	1
0.2	Methodology . . . . .	1
0.3	Business Model . . . . .	4

## 0.1 Introduction

Introduction goes here

## 0.2 Methodology

For the development of our Coach Freelancing Platform, we wanted to choose the software development process that would suit the needs and time constraints of this project. To achieve this, we dedicated an entire meeting to discuss each process we have learned about and list their pros and cons and arrived at the conclusion that we wanted to adopt an agile methodology, since they are the best suited for Software Development. Agile utilize iterative and incremental development. It promotes flexible responses to change and aims to deliver working software frequently, with a preference for shorter time cycles. In software development requirements can change rapidly. Agile is particularly effective, because the project's direction can be changed quickly, but it requires a high level of coordination and communication among team members. During our Software Engineering lectures we learned about many methodologies, here we will go through the most notable ones and argue why we chose to work with the agile framework: Scrum.

### Waterfall Model:

This model is one of the earliest methodologies in engineering, it was and still is really popular outside of Software Engineering. It is based on linear sequential flow,

meaning that any phase in the development process begins only after the previous one has been completed. The model is simple and easy to understand and use, which makes it suitable for projects with well-defined requirements that are unlikely to change, because after the report is done, and the project is planned out ahead, we would have to work according to that plan, even if user-requirements have not been correctly anticipated or have changed. The changing requirements typical in Software Development made this methodology unsuitable for us.

## **Spiral Model:**

The Spiral model combines iterative development with the systematic aspects of the Waterfall model. It allows for incremental refinements through each iteration or spiral. Each spiral starts with a set of goals and ends with the client reviewing the progress, which helps in early detection and reduction of risks. This model is particularly useful for large, complex, and high-risk projects. However, it can be more costly than other methods due to the ongoing evaluation and risk analysis and additionally, the model's complexity necessitates strict adherence to protocols and increased documentation.

## **V-Model:**

The V-Model, or Validation and Verification model, provides a more structured approach which entails a lengthier planning period analogous to the waterfall method of software development. The planning of it must be very reliable as all the successive development phases almost exclusively rely on the previous planning phase. This model is often used in safety critical systems, because is useful for catching defects in early stages, which makes it cheaper and less time-consuming to fix them compared to later in the development process. Given the importance of the ahead of time planning the framework results to be quite inflexible and doesn't take into consideration the possible changing of requirements during the development of the project.

## **Unified Process**

This methodology is structured, iterative, and incremental and it emphasizes risk assessment, architecture, and continuous stakeholder involvement. It achieves these through distinct phases(Inception, Elaboration, Construction, Transition) and iterations which are, unlike Scrum, different length timeboxes, and not emphasized the same. Its values include adaptability, quality focus, and comprehensive documenta-

tion. While the Unified Process is suitable for complex, large-scale projects due to its detailed planning and risk mitigation, it can be expensive and complex, making it less agile and flexible compared to Scrum.

## Agile

We already mentioned why we choose the agile methodology, now we had to choose a framework that utilizes it. We learned about many frameworks, a few of those were frameworks for large scale projects, which we aren't going to go in depth about here, since the magnitude of this project doesn't require it. We learned in depth about XP and Scrum. Since both of these are mostly similar, here we are going to mention a few key features and then argue why scrum is the more suitable for our project.

- Extreme Programming

Extreme programming focuses on customer satisfaction and responding to changing customer requirements through practices like pair programming, test-driven development, and continuous integration. It also encourages frequent releases in short development cycles to improve productivity and introduce checkpoints for new customer requirements and focuses on simplicity, communication, feedback, and courage among the development team.

- Scrum

Scrum is built on the Agile Manifesto and the 12 Agile Principles. It uses fixed-length iterations called Sprints, lasting typically 2-4 weeks. The Scrum Team has defined roles: Product Owner, Scrum Master, Development Team. In and around the sprints there are structured events: Sprint Planning, Daily Scrum, Sprint Review, Sprint Retrospective. Scrum helps teams deliver value incrementally in a collaborative way. It has the artifacts Product Backlog, Sprint Backlog and Increment. It is important to maintain a prioritized Product Backlog to manage changing requirements and ensure that the team works on the most valuable tasks.

We chose scrum, because it is ideal for a small software startup as it fosters rapid development and adaptation to changing market demands. Scrum's lightweight framework is easy to understand and implement, allowing our team to stay focused, aligned, and responsive to our customers' feedback and business priorities. It promotes team collaboration and self-organization, and ensures regular delivery of valuable software increments, which values are important to almost all Software Developments.

## 0.3 Business Model

### Chapter 5: Business Model

By analyzing industry trends and identifying opportunities we plan to tailor our business model. This tailored approach aims to not only effectively reach our target audience but also strategically position our offerings amidst competitors. Based on personal experience and the today's world full of digital content and enormous social media influence, we came to the conclusion that there would not be better way to attract customers than doing so digitally.

Our Coaching Freelancing Platform aims to solve a key problem that the majority of the fitness enthusiasts and health instructors have to face when it comes to online coaching – the absence of communication, the lack of easy-access workouts and direct contact with your trainer and the difficulty of effortlessly finding trainees. Our product provides digital environment where both coaches and trainees benefit from each other. Trainers can create their own profile listing their expertise, awards and pictures, diplomas and certificates, years of experience, area of training, result of previous clients and their feedback. If they already have a greater number of followers, it be extremely easy for them to find clients through the platform. If not, they can promote themselves and rise above other of their colleagues. Additionally, building clients' workouts won't be so time consuming and demanding for them as we provide a workout builder with variety of exercise to choose from, set reps and sets and place it in a monthly calendar. Fitness enthusiasts on the other hand will have the easiest way to find someone to help them achieve their health goals. With the ability to filter area of training, years of experience, testimonies, picture-based result, etc., their decision will be trouble-free. Furthermore, we have solved the inability to communicate with your online coach by providing private one-on-one chatting channels between the trainee and the trainer. In those channels, both sides will be provided with statistics of the progress of the customer.

Our business plan include multiple revenue sources to support. Our primary source of income will come from transactional commissions, where we take a percentage from each coaching transaction made through our platform. Additionally, we will explore revenue streams by letting different sporting brands to promotes their products on our application. Moreover, leveraging a comprehensive database featuring user preferences, gym performance metrics, sleep and dietary habits, as well as body measurements, stands as a pivotal asset.

Having said the wide scope of our web-application, we will have a lot of research and development expenses, programmers' wages. Additionally, a budget for marketing and advertising fees has to be allocated as well.

We are going to have retail partnership with sports well-known brands who have big influence over the health community. Trainers will have easy access to their accounts through a web-site whereas trainees will be able to download the application from the App Store and Google Play Store.

The key task our team needs to face is the development and deployment of the Coach Freelancing Platform. This includes various stages, including planning, software design and architecture, coding and programming, testing and quality assurance, user interface and experience design, as well as implementation and ongoing maintenance. Additionally, our team will focus on providing continuous support to ensure the platform operates smoothly, addressing any technical issues promptly, and incorporating user feedback to enhance the overall user experience.

Two essential resources critical to building and operating our software are a robust database infrastructure and the intellectual property associated with our application. A well-structured database is fundamental for storing and managing user data, coach profiles, workout routines, and other essential information efficiently. Furthermore, safeguarding the intellectual property rights to our application ensures that our innovative features and functionalities are protected, preventing unauthorized use or replication by competitors.

Two strategic partnerships crucial to our business model are with payment processing platform Stripe and handling transactions efficiently. Integrating with Stripe enables seamless and secure payment transactions on our platform, enhancing user trust and facilitating monetization through commissions. Additionally, partnering with transaction handling services ensures reliable and efficient processing of coaching session payments, reducing friction for users and coaches alike while streamlining our revenue generation process. These partnerships play a vital role in optimizing the financial operations of our Coaching Freelancing Platform and enhancing the overall user experience.

Health and fitness is an industry that is extremely dynamic and develops faster than ever. Therefore agility and adaptability are very important for our business model. By closely monitoring market trends, technological advancements, and user feedback enables us to quickly adapt and improve our platform, adding new features, improving user experience, as well as staying ahead of the competition. Our agile way of working guarantees that Coaching Freelancing Platform adapts to diversifying customer tendencies as well as changing the market situation, thus, ensuring the project's success in the long-term perspective.