|  |  |
| --- | --- |
| pROJECT pLAN | Project plan for an individual assignment in the second year S3-CB-S3-CB05 |

Table of Contents

[THE CLIENT 2](#_Toc176784086)

[TEAM 2](#_Toc176784087)

[WAY OF WORKING 3](#_Toc176784088)

[CURRENT SITUATION 3](#_Toc176784089)

[PROBLEM DESCRIPTION 3](#_Toc176784090)

[PROJECT GOAL 3](#_Toc176784091)

[DELIVERABLES 4](#_Toc176784092)

[NON-DELIVERABLES 4](#_Toc176784093)

[Testing Strategy 5](#_Toc176784094)

[RISK MANAGEMENT 6](#_Toc176784095)

[CONSTRAINTS 7](#_Toc176784096)

[ProjectMilestones 7](#_Toc176784097)

[PHASING 8](#_Toc176784098)

## 

## THE CLIENT

* A client of the project is represented by Maja Pesic and Marcel Boelaars, who are teachers at Fontis University of Applied Sciences.
* Contact person: Maja Pesic
* Email address: m.pesic@fontys.nl.
* Contact person: Marcel Boelaars
* Email address: m.boelaars@fontys.nl

## TEAM

* Our team is represented by Atanas Dimitrov (a student at Fontys University), who will take care of the overall preparation of the project, together with its documentation.
* Email address: [atanas.dimitrov@student.fontys.nl](mailto:atanas.dimitrov@student.fontys.nl)

## WAY OF WORKING

* The project's approach involves gradually integrating new concepts and knowledge gained throughout my learning process. As I continue to learn, I will consistently apply newfound knowledge to enhance the project. Regular meetings with my teachers will provide valuable guidance and feedback, contributing to the improvement of both the work process and the overall quality of the final product.

## CURRENT SITUATION

* The current landscape of fitness and diet websites is highly competitive, with numerous platforms offering users access to workout and nutrition plans. However, many of these platforms provide generic, one-size-fits-all solutions that fail to address individual needs. Modern trends are shifting toward personalized, data-driven plans that cater to specific user goals, body types, and preferences. Users now expect more interactive tools that offer adjusted workout routines and diet recommendations based on factors like their fitness level, goals, and nutritional requirements, enhancing both effectiveness and engagement.

## PROBLEM DESCRIPTION

* Users often face challenges finding a platform that combines both personalized workout plans and adjusted diet suggestions in a single, user-friendly interface. Existing solutions tend to either focus exclusively on fitness or nutrition, lacking comprehensive integration. Additionally, many platforms fail to offer adequate tracking features, leaving users without clear progress monitoring tools. This results in a disjointed experience, forcing users to juggle multiple apps or websites to manage their fitness and diet needs, limiting their overall success and engagement.

## PROJECT GOAL

* The primary objective of the project is to create a user-friendly website that provides personalized fitness and diet plans adjusted to individual needs and preferences. The platform will use user data, such as fitness goals, workout preferences, and dietary requirements, to offer customized recommendations for both exercise and nutrition.
* **The secondary objectives include:**
  + Enabling users to track their progress through features like photo uploads and notes.
  + Allowing trainers to manage their clients by creating workout plans, viewing client profiles, and monitoring their progress.

## DELIVERABLES

* **For Users:**
  + User account and profile.
  + Personalized workout and diet suggestions based on preferences.
  + Options for choosing the fitness instructor.
  + Progress tracking system (pictures and notes).
* **For Trainers:**
  + Profile creation.
  + Set profile card for advertise themselves.
  + Ability to build customized workout plans.
  + Access to user progress.
* **Technical Deliverables:**
  + Fully responsive website (cross-browser, mobile-friendly).
  + Backend database to manage users, trainers, and plans.
  + Comprehensive test report.
* **Documentation:**
  + Project Plan: Detailed timeline and phase breakdown of the project milestones and deliverables.
  + Design document: detailed technical specification that outlines the architecture, components, and functionality of a system or project.

## NON-DELIVERABLES

* **Continuous Platform Maintenance Post-Launch**: Ongoing maintenance and support will be contracted separately outside of the project scope.
* **Marketing:** Content creation for trainers and advertising is not part of the scope.
* **Mobile Application:** The development of a mobile version of the platform is not included in this project phase.

## Testing Strategy

To ensure that the fitness and diet website functions optimally and meets user expectations, a multi-layered testing approach will be adopted:

* Unit Testing: Each website component, such as user registration, login, and workout/diet suggestion algorithms, will undergo unit testing. This ensures that individual features work correctly in isolation.
  + Goal: Achieve 80% code coverage for critical functions.
* Integration Testing: This level will verify that the various components of the system (user profiles, workout plans, diet plans, and progress tracking) integrate seamlessly with the back-end, database, and third-party APIs.
  + Goal: Ensure all system integrations function at 95% success rates across different workflows, including user-to-trainer communication and progress updates.
* System Testing: The entire system will be tested as a whole to ensure the platform delivers a smooth experience from account creation to plan generation and tracking.
  + Goal: Confirm the entire user journey works end-to-end, covering login, plan selection, progress tracking, and trainer interactions.
* Acceptance Testing: The final review with stakeholders, including trainers and test users, to ensure the system meets the original project goals and client expectations. This will include user feedback and usability testing.
  + Goal: Achieve full acceptance of key project deliverables before final deployment.
* Code Quality and Security Testing: Use tools like Sonarqube to check for code vulnerabilities, maintainability, and code quality issues such as logic flaws, security risks, and performance bottlenecks.
  + Goal: Ensure that code meets quality standards with minimal critical vulnerabilities.

## RISK MANAGEMENT

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Risk | Preventive Measures | Resp Action | Probability | Impact |
| If someone is sick | Regular health checks, balanced workload | If I get sick it won't be a problem as I am organized and will easily be able to make a plan to follow to achieve my goals. | High | Medium |
| If the code is broken | Frequent code reviews, use of version control | I will revert to the previous stable version from my GitLab repository. | Low | High |
| If my GitLab repository is deleted | Regular backups to external storage | I will upload the latest version of the code from my laptop. Additionally, I will contact GitLab support to investigate the possibility of recovering my repository. | Low | High |
| If I have problems with the implementation | Scheduled maintenance, having backup devices | First, I will try to find help on the Internet. If that doesn't help, I will ask my teachers for help. | Low | High |
| If there is a problem with the technique I am working with. | Regular backups to external storage | I will ask to borrow a laptop from ISSD in Fontys. I will then link it to my GIT account and continue working on the project. | Medium | Medium |
| If I don't have time for my results | A pre-prepared timeline, with clear deadlines and milestones | I'll create a table categorizing them into 'must have,' 'should have,' and 'could have' to prioritize effectively. | Low | Medium |

## CONSTRAINTS

* Budget: $ 0;
* Workforce: 1 workers;
* Initiation: 06 September 2024;
* Deadline: 17 January 2025;
* Use limited to Java, HTML/CSS, JavaScript, React, MSSQL;

## [Project](#_Project_Milestones)[Milestones](#_Project_Milestones)

* **Sprint 1: Documentation** 
  + Deliverable: Ideation Document, Project Plan
  + Due Date: 20 September 2024
* **Sprint 2: Design document** 
  + Deliverable: Design document, Backend, Initial Frontend setup
  + Due Date: 11 October 2024
* **Sprint 3: Design document version 2** 
  + Deliverable: Design document, Initial Backend to Database setup, SonarQube
  + Due Date: 08 November 2024
* **Sprint 4: UX feedback**
  + Deliverable: UX feedback report, Design document, Authentication and Authorization implementation, Continuous Integration and Sonarqube
  + Due Date: 29 November 2024
* **Sprint 5: Final design document**
  + Deliverable: Final design document, Security report, Websockets feature, MVP features implementation, Continuous Integration and Sonarqube
  + Due Date: 20 December 2024
* **Sprint 6: Final individual track product**
  + Deliverable: Final individual track product, Continuous Integration and Sonarqube, Continuous Delivery, A web performance review document
  + Due Date: 17 January 2025

## PHASING

**1. Planning and Preparation**

* Define project goals, objectives, and requirements: Identify the purpose and expected outcome of the project.
* Create a project plan: Outline tasks, timelines, and milestones, ensuring each task has clear deliverables.
* Set up tools: Establish GIT for version control and select project management tools (e.g., Jira, Trello).
* Stakeholder feedback: Present the plan to stakeholders (project tutor, client) for feedback and make adjustments based on input.

**2. Research and Initial Learning**

* Research technologies: Identify tools, frameworks, and methodologies relevant to the project.
* Apply study materials: Review relevant Fontys University study content and incorporate appropriate concepts.
* Identify challenges: Highlight areas where additional learning may be needed.
* Basic prototype development: Start working on a basic version of the app that includes core functionalities (e.g., basic UI, database integration).
* First feedback session: After initial research and prototype, seek feedback from stakeholders and adjust the project as necessary.

**3. Initial Submission & Iteration**

* Submit basic version: A basic working version of the web app with fundamental functionalities, ensuring a functional demo (e.g., login, data retrieval).
* Feedback on initial submission: Gather feedback from stakeholders and incorporate any necessary changes or improvements.

**4. Development of Core Features**

* Implement main features: Begin development of the remaining key features and functionalities based on project requirements.
* Regular testing and debugging: Continuously test features to ensure functionality and identify bugs early.
* Weekly feedback loop: Present progress to stakeholders at the end of each week for feedback and adjust development plans accordingly.

**5. Integration and Testing**

* Database integration: Ensure smooth integration between the web application and the database.
* Conduct thorough testing: Perform unit, integration, and user acceptance testing to find and resolve any bugs.
* Stakeholder feedback: After testing, share results with stakeholders, gather feedback, and refine the system as needed.
* Iterate based on testing: Make improvements to both functionality and performance based on feedback and test results.

**6. Feedback and Refinement**

* Final round of feedback: Present the near-final version to stakeholders and gather final feedback on performance, UI/UX, and functionality.
* Incorporate refinements: Make the final tweaks and adjustments to the web app based on this feedback.

**7. Final Submission and Deployment**

* Final testing and quality assurance: Conduct a final review of the web app, ensuring that all functionalities work as expected.
* Prepare for deployment: Get the app ready for submission and deployment, ensuring that it meets the submission criteria.
* Submit final version: Upload the final version of the application to the assignment section in Canvas for evaluation.