POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Title

Tesi di Laurea Magistrale in
Xxxxxxx Engineering - Ingegneria Xxxxxxx

Author: **Name Surname**

Student ID: 000000
Advisor: Prof. Name Surname
Co-advisors: Name Surname, Name Surname
Academic Year: 20XX-XX

Abstract

Here goes the Abstract in English of your thesis followed by a list of keywords. The Abstract is a concise summary of the content of the thesis (single page of text) and a guide to the most important contributions included in your thesis. The Abstract is the very last thing you write. It should be a self-contained text and should be clear to someone who hasn't (yet) read the whole manuscript. The Abstract should contain the answers to the main scientific questions that have been addressed in your thesis. It needs to summarize the adopted motivations and the adopted methodological approach as well as the findings of your work and their relevance and impact. The Abstract is the part appearing in the record of your thesis inside POLITesi, the Digital Archive of PhD and Master Theses (Laurea Magistrale) of Politecnico di Milano. The Abstract will be followed by a list of four to six keywords. Keywords are a tool to help indexers and search engines to find relevant documents. To be relevant and effective, keywords must be chosen carefully. They should represent the content of your work and be specific to your field or sub-field. Keywords may be a single word or two to four words.

Keywords: here, the keywords, of your thesis

Contents

Abstract i

Contents iii

1 Introduction 1

1.1 Group Information . . . 1
1.2 Project Information . . . 1
1.3 Work Breakdown . . . 1

2 Documentation 3

2.1 Chosen Theme . . . 3
2.2 Technological Choices . . . 3
2.3 Project Structure . . . 4
2.3.1 Links/Pages Structure . . . 5
2.3.2 Available Server Endpoints . . . 5
2.4 Custom Types . . . 6
2.5 Custom Components . . . 6
2.5.1 Buttons . . . 6
2.5.2 Cards . . . 6
2.5.3 Carousels . . . 6
2.5.4 Containers . . . 6
2.5.5 Slides . . . 6
2.5.6 Other Components . . . 6
2.6 Extra Modules . . . 6
2.7 External Libraries . . . 6

3 Extras 7

1 | Introduction

1.1. Group Information

Group Name: HyperDaGiaZeySe

Group Composition:

Group Composition

Name Surname	Person Code
Zeynep Erbaysal	11035715
Sebastian Perea	10986638
Daniel Ruiz	10988760
Giacomo Scampini	10764570

1.2. Project Information

Website: `https://lotus-haven.vercel.app/`
GitHub Repository: `https://github.com/sebaxe07/LotusHaven`

1.3. Work Breakdown

Throughout the project, we aimed to maintain a balanced and collaborative workflow, ensuring that each team member could contribute and learn from different aspects of the development process. As the project progressed, we naturally divided responsibilities to leverage individual strengths and improve efficiency. Sebastian Perea and Daniel Ruiz primarily focused on the core development and implementation of the application, while Zeynep Erbaysal and Giacomo Scampini concentrated on the website's styling and user interface design. Despite this division, all members participated in discussions and contributed ideas to both the technical and design components, resulting in a cohesive and well-rounded final product.

2 | Documentation

2.1. Chosen Theme

This project presents the design and development of a yoga center website focused on promoting wellness, clarity, and connection. The website architecture is built around key entities such as instructors, activity types, and scheduled sessions, with main navigational pages including the homepage, About Us, and Contact Us. The design process included structured content organization, detailed wireframes with design rationale, and realistic user interaction scenarios to ensure intuitive navigation. The underlying database schema supports efficient data flow and user interaction, providing a cohesive and user-friendly experience for visitors seeking information about yoga activities and instructors.

2.2. Technological Choices

- **Hosting: Vercel** – The website is deployed and hosted on Vercel, a cloud platform optimized for frontend frameworks and static sites. Vercel provides seamless integration with GitHub for continuous deployment, fast global CDN, and automatic SSL, ensuring high availability and performance.
- **Database: Supabase** – Supabase is used as the backend database solution. It offers a scalable, open-source alternative to Firebase, providing a PostgreSQL database, authentication, and real-time capabilities. Supabase enables secure storage and efficient retrieval of data related to instructors, activities, and sessions.
- **Rendering Mode: Server-Side Rendering (SSR)** – The project is configured to use Nuxt's server-side rendering mode, as specified in the configuration file. All routes are rendered on the server by default, providing improved SEO, performance, and dynamic content handling. This ensures that users receive fully rendered pages from the server, with client-side hydration for interactivity.
- **Main Framework: Nuxt** – The application is built using Nuxt, a powerful Vue.js framework for building modern web applications. Nuxt provides features such as

file-based routing, automatic code splitting, and an intuitive module system, streamlining development and improving maintainability.

- **Programming Language: TypeScript** – TypeScript is used throughout the codebase to provide static typing, improved code quality, and better developer tooling. TypeScript helps catch errors early and enhances the maintainability of the project.
- **Component-Based Architecture** – The project is organized using a modular, component-based structure, with reusable UI components for buttons, cards, carousels, and more. This promotes code reuse and simplifies updates and testing.
- **Version Control: GitHub** – Source code is managed using Git and hosted on GitHub, enabling collaboration, version tracking, and integration with CI/CD pipelines.

2.3. Project Structure

2.3.1. Links/Pages Structure

Website Navigation Structure

Page	URL	Description
Home	/	Landing page with hero section, featured activities, teacher highlights, and call-to-action sections providing an overview of the yoga studio.
Highlights	/highlights	Displays featured yoga classes and special activities, including a prominently featured activity and a carousel of additional highlighted options.
Activities	/activities	Lists all available yoga classes with search functionality and filtering options. Users can browse and find detailed information about each activity.
Activity Detail	/activity/[id]	Shows detailed information about a specific yoga class, including description, difficulty level, duration, and teacher information.
Teachers	/teachers	Displays all yoga instructors with their specialties and a brief introduction. Users can click to view detailed profiles.
Teacher Profile	/teacher/[id]	Provides comprehensive information about a specific teacher, including their bio, expertise, contact information, and scheduled classes.
About Us	/about	Contains information about the yoga studio philosophy, mission, location, operating hours, and other facility details.
Contact Us	/contact	Features contact information, staff directory, location map, and a contact form for inquiries and class registrations.

Table 2.1: Website navigation structure showing main pages and their purposes

2.3.2. Available Server Endpoints

2.4. Custom Types

2.5. Custom Components

2.5.1. Buttons

2.5.2. Cards

2.5.3. Carousels

2.5.4. Containers

2.5.5. Slides

2.5.6. Other Components

2.6. Extra Modules

2.7. External Libraries

3 | Extras