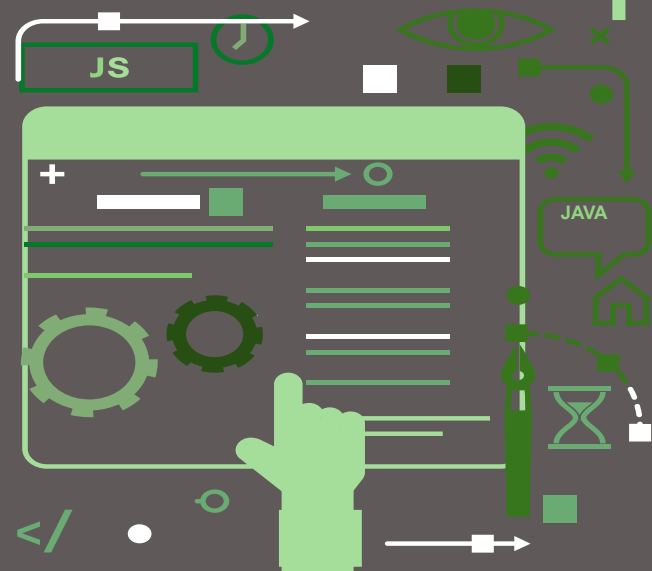




# WEB APPLICATION

## Fitness Tracker



### CycleK Team:

Abedini Kimia  
Boscolo Bacheto Martina  
Cocco Alessio

Munerotto Giacomo  
Tomaiuoli Marco  
Trevisiol Riccardo

# TABLE OF CONTENTS

---

## About the Project

Brief introduction of the Project

1



## Project Goals

Project main features

2



## Backend/Database

Entities involved and data manipulation.

3



## Authentication

Authentication with JWT token and cookies.

4



## Frontend

User Interface design and development.

5



## External libraries

Bootstrap and chart.js for a responsive and modern style.

6









# 1

## ABOUT THE PROJECT



Our web application defines a **Fitness Tracker**, capable of managing different gym related aspects:

In particular, our Web Application is divided in six main parts:

- User Management 
- Statistics Management 
- Social Network Management 
- Diets Management 
- Meals Management 
- Exercise Management 

## 2

## PROJECT GOALS



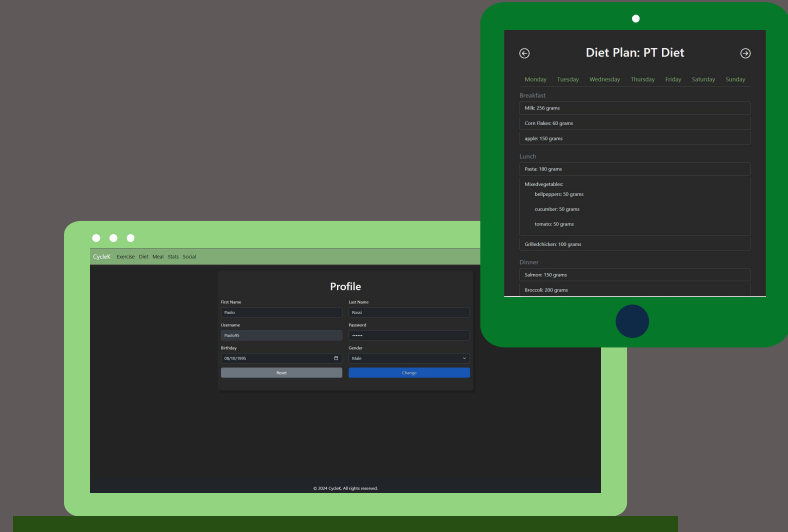
Manage his profile



Visualize his progress

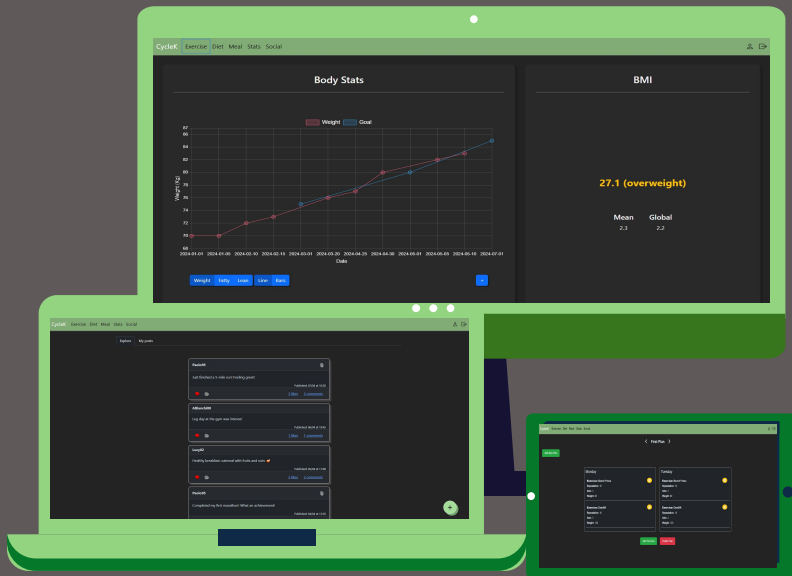


Visualize/create his own diet



## 2

## PROJECT GOALS



Visualize/create his own  
exercise schedule



Compute nutritional values of  
his meal of the day



Keep in touch with the  
fitness-tracker community



# 3

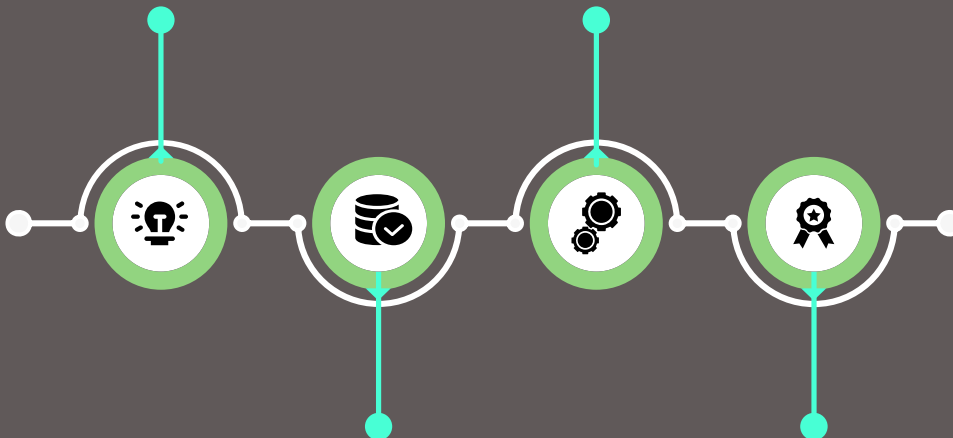
## BACKEND



The **Rest Dispatcher** identify a request and forwards it to the appropriate **Rest Resource**

The **Rest Resource** receives the DAO's Response and write it in JSON format

RECEIVE THE REQUEST



The **Rest Resource** forwards the Request to the appropriate **DAO**

The **Dispatcher** processes the JSON received by the Rest Resource

SEND THE RESPONSE





We decided to assign the **JSON** type to our complex data:

- Easy to handle **complex data**



- **Useful libraries** to handle JSON conversion i.e. JACKSON



- JSON databases are **faster** and have more storage **flexibility**

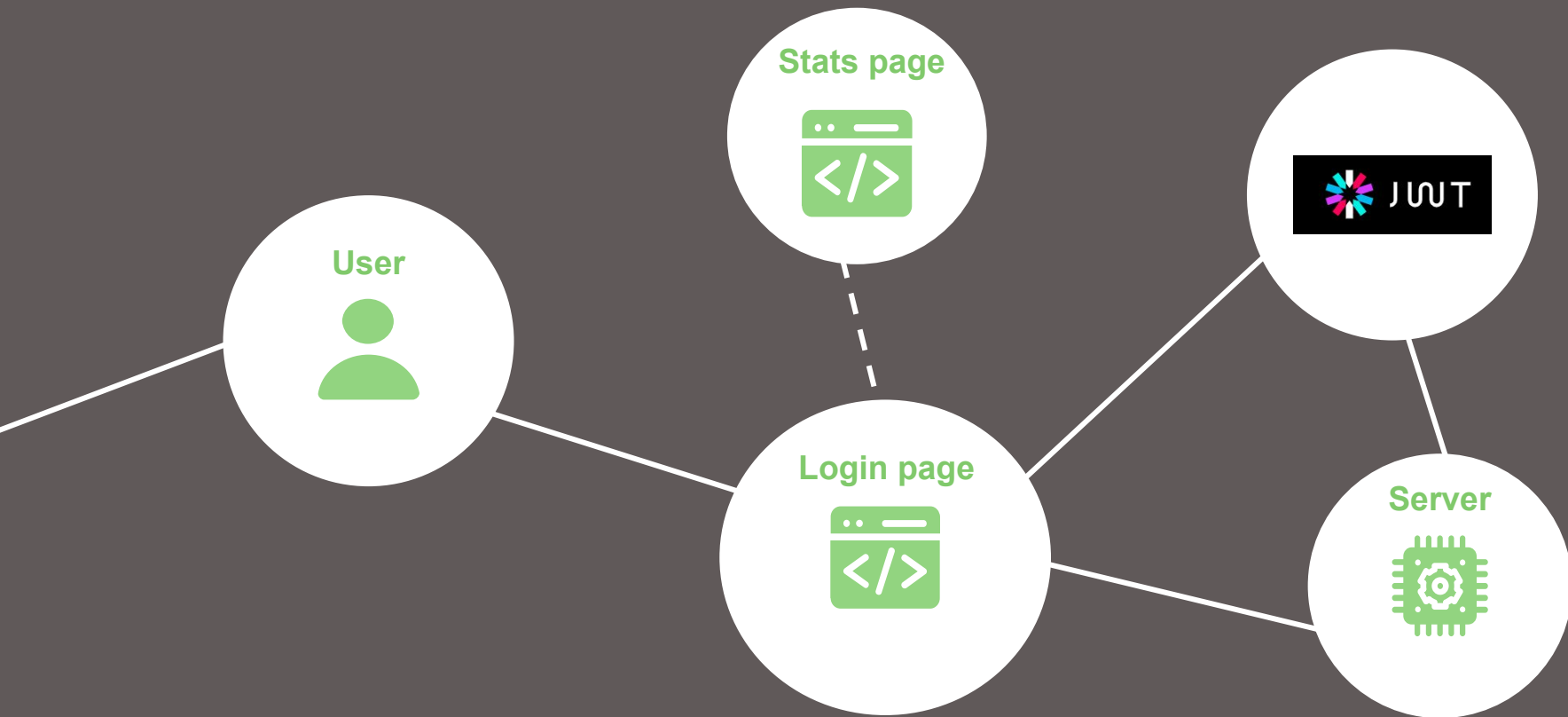


- Both diets and exercises have **lots of data** to be stored



# 4

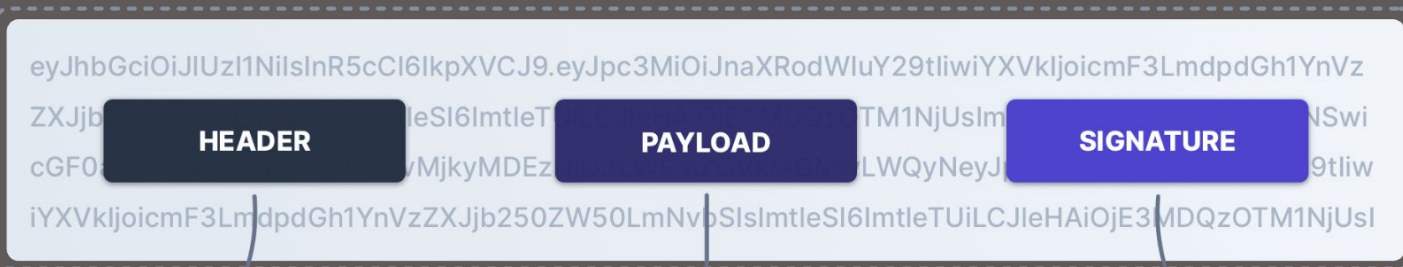
## AUTHENTICATION





## 4

## JSON WEB TOKEN (JWT)



```
{  
  "alg": "HS256"  
}
```

```
{  
  "sub": "1",  
  "exp": 1717755037  
}
```

idUser  
time

```
HMACSHA256(  
  base64UrlEncode(header) + "." +  
  base64UrlEncode(payload),  
  SECRET_KEY  
)
```

☒ secret base64 encoded



Developed using:

- **HTML** for providing the structure and content of the page;
- **CSS** for styling the HTML elements;
- **JavaScript** for adding client-side interactivity and dynamic behavior to the HTML elements.



HTML, CSS, and JavaScript are integrated and managed within **JSP pages** to create dynamic and interactive web content.

# 5

## FRONTEND

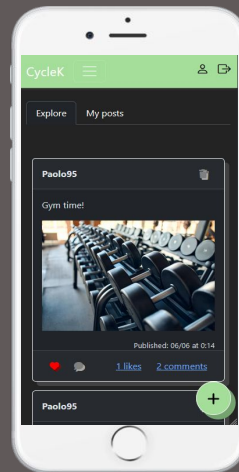


We chose a **dark theme** with **light green** tones, which are featured in the navbar and on some buttons.



There is a **common** style.css file, to be shared among all the pages

Our WebApp is **mobile-friendly**, utilizing the **viewport meta tag** and **CSS media queries** to ensure optimal display and functionality across various device sizes



# 6

## EXTERNAL LIBRARIES

---



**Bootstrap** is a front-end framework for responsive and mobile-first web development using HTML, CSS, and JavaScript.

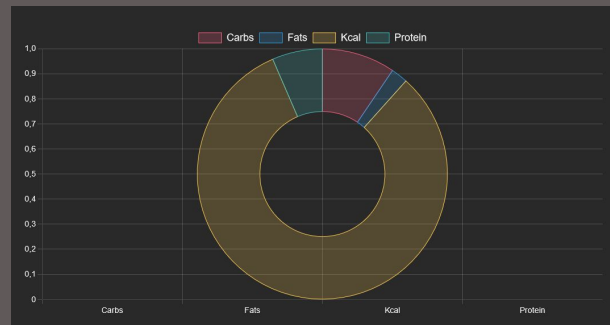
- pre-designed components
- consistent styling

## 6

# EXTERNAL LIBRARIES



**Chart.js** is a JavaScript library for creating interactive and responsive charts in web applications



**Font Awesome** is an icon toolkit to enhance the design and functionality of websites and applications.



**And now we will show our  
demo!**