**Main stages of project development**

0) Create Software Requirement Specification(SRS)

Discuss and negotiate project plan. Create page prototypes. Create full pages list. Create project sitemap. Create description for UI behaviours.

1) Create set of tasks

Main setup(project, code repository, milestone deadlines, etc) Setup blog, live site, staging server. Discuss email notifications provider.

2) Project Start

By the end of this stage, we'll have project with created first working version of database, created different user roles, main functionality like add/edit/delete entities - done. Wrap labels, titles, flash messages to translation functions.

Build working version, without design, without titles, but with multi language setup. A lot of simple pages with forms(later will be combined to one pages). This will be easy to create and easy to test.

Database with data(food, activity) must be imported from start. If not, it can cause some troubles.

3) Middle stage.

By the end will be completed admin and food database logic. After this stage database cannot have complex adjustments. Add meal plans analytics, test with different amount of entered data

Rebuild some functionality, testing complex things, math formulas.

Testing appointments, meal planning.

4) 70% completed stage.

Completed professional side. Payments gateways enabled. Appointments and meal planning functionality.

5) Testing, debugging, bugfixing.

On this stage our main task - try to find bugs, using different ways of testing. And fix it.

Main goals: registrations, food search, meal planning, payments, emails.

6) Add styles to worked well project. Create and apply email templates styles.

9) Bug fixes, speed testing, test on different devices, etc.

10) After launch part. Custom design stage. Created UX by experienced designer

11) Convert design to HTML for each page

12) Implementation HTML to worked project

Add js features, like modals, graphs, animations, inline edit plugins. Remove or hide(read disable on stable version) small unnecessary pages required to forms development, functionality testing.

Project will have a high level of security. For user accounts we'll setup RBAC authentication. system.

**How i'll build this project.**

Firstly i'll create and describe all step-by-step tasks, which we'll need to cover during this project. Don't forget to ask and get answers from client.

Create list of all necessary pages.

Describe any page(view) in details(design, elements, functionality, etc.).

Describe all forms, which must be on project.

Describe all email notifications, which must be created on project.

Project will have 3 main roles with different dashboards, pages: Patient, Professional(User), Admin

**Forms**

Users(professional):

~ quick registration

~ forgot password form

~ contact us form

* Name
* Email
* Subject
* Message

~ login form

* Email
* Password

~ change subscription plan

~ update photo profile

~ update/Add workplaces/Search

~ system preferences

~ personal details

~ subscription information

~ checkout card information

~ billing information

~ schedule new appointment

~ edit appointment modal form

~ add patient(short)

~ patient information

~ appointment information

~ personal and social history

~ medical history

~ goals form

~ dietary history

~ measurements(Anthropometric Data/Analytical Data/Body Composition)

~ Planning

* Patient information
* Calculations
* Macronutrients
* Nutrients qualification
* Duration
* Create meal plan version
* Import meal plan
* Add food
* Edit food
* Duplicate food
* Add common measure food

~ Physical activity / add / edit search / some notes

~ Checkout form

**Email templates(additional document)**

Other emails:

Appointment scheduled

Changes in appointment schedule( update / cancel / next appointment starting soon )

Meal plan

**Pages(views) list**

**Front pages**

* Home
* Terms
* Privacy
* Contact
* FAQ
* Refund policy
* Blog
* Sign in
* Sign up
* Retrieve Password
* Support
* Blog Article
* 404 page
* 500 page and other error pages

**Admin pages**

* Languages management
* Blog(editor)
* User management, patient management
* Edit
* Add ability to see users, related to coach
* Orders
* Food Management
* Support, contact messages(maybe)
* Website content and settings
* Main dashboard page, with necessary information

**Main API calls**

* Meal Plans GET/ADD/EDIT
* Patient Info GET
* Patient Measurements GET
* Food
* Appointments GET
* Goals GET/ADD
* Calendar GET/ADD/CANCEL

**Professional(user) pages**

* Dashboard
* Profile
* Subscription

- Checkout

- Thank you page - after payment

* Logout
* Register patient
* Schedule Appointment

- Add appointment

- Edit appointment

- Cancel

* Workspaces(find this feature on latest blog post)

- Manage(Create/Edit/View)

**Patient(Under User) pages**

* Add
* Manage(Edit/Delete)

- Information (complex)

- Measurements (complex)

- Planning (complex)

- Meals (complex)

- Recommendation (complex)

- Analysis (complex)

- Appointments (complex)

- Next appointment

- Finish appointment

Patient & appointments info | Measurements | Medical info | Goal management | Personal history | Dietary History

**Foods(Under User) pages**

* Manage( Add/Edit/View )
* Duplicate food
* Search
* Filtration
* Pagination

**Meal Plans(Under User) pages**

* Add template
* Manage(View/Edit/Remove)
* Versions(Different plans for weekdays)
* Import
* Manage Meals(Edit, Add foods)

**DB entities**

* Appointment
* Patients
* Meal plans
* Activities
* Foods(Group, Facts, Units, Measurements )
* Measurements
* Orders

**Graph and widgets**

Distribution(meal plan analytics) & graphs with different logic of calculation

New patients

General info

Age distribution

Appointments volume

Patient Progress info

Energy / Macronutrients / Micronutrients

**Complex Architecture Logic**

*Meal plan logic*

Meal plan template management: save information, create id. When you save template - you have meal\_plan\_id but without patient id.

*Physical activity database with data about calories, time, etc.*

*Food Database tables.*

*Users(add parent\_id field)*

If admin create professional account - parent\_id = admin ID

When professional create patient account - parent\_id = patient\_id

*Companies(workspaces)*

Only professional can have companies(right now)

Can have more than one company

**Questions**

* Do you need to have feature "food database"?
* Do you need to have feature to help users track weight loss?
* Do you need to have feature to have recipes?
* Do you need fitness trackers integrations?
* Do you need alerts or timers?
* Do you need to have translations?
* How many languages you need to support?
* Do you need to setup newsletter?
* Can one patient belong to a few different doctors?
* Can different doctors belong to one workspace?
* If one doctor is ill, can he move patient appointment to another doctor?
* If doctor create a new patient, he’ll receive a notification? Or notification send only when appointment scheduled?
* Create inner nutrium email functionality. What is the main reason for this functionality? We need it for login, or for sending notifications, meal plans, etc.
* Import food from other meal plan - How it works if we haven’t any meals created yet?
* “Prediction equation” - I think I need more information about this
* Can you give me more details for recommendation logic. This is default formula, like *weight = height - 100* ? Create ability to add unique calculations
* For each graph we need to understand logic
* Can I delete patient? All stored patient data erased?
* If my account was suspended - do we freeze all or delete data?
* Do we need to have unsubscribe, upgrade, downgrade functionality for subscription plans?
* Default data(SR27, USA) - do we need to import data to database? Food database - we have it before start?
* I need to analyze recommendation logic more.
* Discuss multilanguage(i see Portuguese language): list of languages, how to manage languages.
* Enable cookies?
* Order history section on profession subscription page with payment card history?
* If I have allergies(patient user role) and this is stored in patient info. If professional make a mistake and add food with allergens. Do we need to have notifications(warnings)?
* When appointment was successfully scheduled - do we need to send email for professional mail address, after appointment submit? Integration with Google Calendar, iCal, Yahoo Calendar?

**Project management tasks**

get emails of all project members.

setup accounts on bitbucket if needed.

Choose project management software

Setup time logging

Setup wiki pages

Setup credentials page

Create page with all page links for testing

Create all tasks. Split tasks on sprints

Local project setup

Setup project on client environment

**Testing**

Test email notifications

Test payments on livesite

Test each form(required fields)

**Features**

BMI Calculator

Calorie Counter

Dieter Assessment

Diet nutrition facts

Use AI for helping build nice meal plans

**Get info**

* Get content for terms and privacy
* Faq section. Setup a default questions and complete it with answers.
* Content for 404, 500 or use default templates
* Homepage. Content, get testimonials list, contact us - get email, get links to social networks, get information from intercom: code, API key, etc.
* Create PDF with Meal plan information - discuss what information need to be displayed
* Discuss email templates: design, information. Multilanguage emails
* Blog(I see that ghost was setup, but default questions, like):
* how client can manage posts, editor logins,
* default blog functionality like comments, articles, categories.
* include SEO
* authors support

This 'blog related questions' not important if you use ghost, but if we go from idea stage - they need to be answered.

* Get custom messages for validation errors, flash message - if needed.
* Access to hosting

**Notes**

Build project with keeping in mind future integration with app.

Using a right approach while architecture created - will save time on later project stages.

Right now design looks minimalistic and cool, but it not help user to enter data.

I’ll propose create and set data for 10-30 patients and create a different meal plans. After this you’ll understand how to improve UX, maybe simplify for the end user.

Country - setup an address book database extension.

**Ideas**

I propose to import data from Apple iHealth or Apple Watches.

Add to settings page options for setting default appointment time(for example 1 hour).

**default** appointment length: 1 hour

**custom1** appointment length: 30min

**first\_appointment**: 2 hours

This can simplify for appointment creation.

Also we need to set a working hours(work schedule) restrictions.

Rules like: cannot create appointment after 5pm, appointment cannot finish at 6pm, cannot create appointment for 10pm

For App - create API(develop architecture with API in mind, can save money(read time) later)

Create landing page for Application, with download buttons, screens, benefits, features.

**Issue**: on checkout page i can add month with value 19