

Mobile Navigation

Θα προσπαθήσουμε να μετασχηματίσουμε την μπάρα πλοήγησης σε προσαρμοστική. Στο breakpoint που ορίσαμε πριν δηλ. στα 940px, θα ήταν καλύτερο η μπάρα πλοήγησης να αποκρύπτεται και στη θέση της να εμφανίζεται ένα κουμπί όπως το πιο κάτω.



Ο χρήστης πατά στο κουμπί προκειμένου να εμφανιστεί το μενού πλοήγησης. Το μενού εμφανίζεται και η μορφή του κουμπιού αλλάζει όπως πιο κάτω. Όταν ο χρήστης πατήσει στο κουμπί το μενού πλοήγησης κλείνει.



Θα ξεκινήσω τοποθετώντας το πρώτο κουμπί στον κώδικα της σελίδας **index.html** στην περιοχή αμέσως μετά την μπάρα πλοήγησης.

Στο αρχείο **Index.html** εντοπίζω τον κώδικα της μπάρας στο εσωτερικό της διπλής ετικέτας **<nav></nav>** και μετά την ετικέτα τέλους αυτής προσθέτω τον επόμενο κώδικα:

```
<header>

<nav>
  <ul class="nav-list">
    <li><a class="btn-menu" href="#">How it works</a></li>
    <li><a class="btn-menu" href="#">Meals</a></li>
    <li><a class="btn-menu" href="#">Testimonials</a></li>
    <li><a class="btn-menu" href="#">Pricing</a></li>
    <li><a class="btn-menu" href="#">Try for free</a></li>
  </ul>
</nav>
<button class="btn-mobile-nav">

</button>
</header>
```

Στη σελίδα με τα ion icons αναζητώ **menu**. Πατώ στο κουμπί με τις 3 γραμμές για να το επιλέξω και στη συνέχεια αντιγράφω τον κώδικα του κουμπιού και τον επικολλώ μέσα στη διπλή ετικέτα `<button> </button>`.

```
<button class="btn-mobile-nav">
  <ion-icon name="menu-outline"></ion-icon>
</button>
```

Ο κώδικας εμφανίζει μέσα στη σελίδα το πιο κάτω κουμπί στα δεξιά της μπάρας



Στη συνέχεια προσθέτουμε στο εικονίδιο την κλάση **icon-mobile-nav** προκειμένου να μπορούμε να το επεξεργαστούμε μέσω των CSS.

```
<button class="btn-mobile-nav">
  <ion-icon class="icon-mobile-nav" name="menu-outline"></ion-icon>
</button>
```

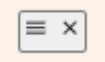
Θα χρειαστούμε και την άλλη μορφή του κουμπιού που θα πατάει ο χρήστης για να κλείσει το μενού πλοήγησης. Στη σελίδα με τα ion icons αναζητώ **close**.



Πατώ στο κουμπί για να το επιλέξω και στη συνέχεια αντιγράφω τον κώδικα του κουμπιού και τον επικολλώ ακριβώς από κάτω από τον κώδικα του προηγούμενου. επίσης το συνδυάζω με την ίδια κλάση **icon-mobile-nav**.

```
<button class="btn-mobile-nav">
  <ion-icon class="icon-mobile-nav" name="menu-outline"></ion-icon>
  <ion-icon class="icon-mobile-nav" name="close-outline"></ion-icon>
</button>
```

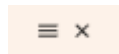
Σώζω και βλέπω τις αλλαγές. Τα δύο κουμπιά εμφανίζονται δίπλα δίπλα μετά την μπάρα πλοήγησης.



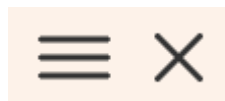
Στο αρχείο **style.css** και στο τέλος των κανόνων που αφορούν την HEADER SECTION προσθέτω τους επόμενους κανόνες.

```
.btn-mobile-nav {  
  border: none;  
  background: none;  
  cursor: pointer;  
}  
.icon-mobile-nav {  
  width: 4.8rem;  
  height: 4.8rem;  
  color: #333;  
}
```

Ο πρώτος στοχεύει στην ετικέτα **button** και αφαιρεί το προκαθορισμένο περίγραμμα που τοποθετεί ο browser και το γκρι φόντο. Επίσης αλλάζει τη μορφή του δείκτη του ποντικιού σε χεράκι όταν ο χρήστης διέρχεται πάνω από την περιοχή αυτή. Το κουμπί παίρνει την πιο κάτω μορφή.



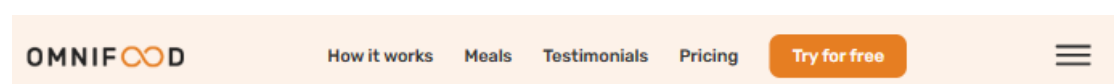
Ο δεύτερος κανόνας στοχεύει στα εικονίδια που περιέχονται μέσα στο κουμπί. Αυξάνει τις διαστάσεις τους δίνοντάς τους πλάτος και ύψος 48px και αλλάζει το χρώμα τους σε σκούρο γρι.



Στη συνέχεια προσθέτουμε τον επόμενο κανόνα με τον οποίο επιλέγουμε ανάμεσα στα δύο στοιχεία που φέρουν την κλάση **icon-mobile-nav** εκείνο με την παράμετρο **name="close-outline"** και το κάνουμε αόρατο. Προσέξτε τον επιλογέα που χρησιμοποιώ σ' αυτή την περίπτωση.

```
.icon-mobile-nav[name="close-outline"] {  
  display: none;  
}
```

Σώζω και βλέπω τις αλλαγές. Το κουμπί **close** εξαφανίζεται και στο δεξί άκρο της κεφαλίδας βρίσκεται πια μόνο το κουμπί **menu**.



Αργότερα ο χρήστης πατώντας στο κουμπί θα μπορεί να εναλλάσσεται μεταξύ του **menu** και του **close icon**.

Αρχικά το στοιχείο button πρέπει να είναι **αόρατο**. Γι' αυτό στον προηγούμενο κανόνα θα προσθέσουμε την πιο κάτω ιδιότητα.

```
.btn-mobile-nav {  
    border: none;  
    background: none;  
    cursor: pointer;  
    display: none;  
}
```

Για διαστάσεις οθόνης από **940px και κάτω** το κουμπί πρέπει να γίνεται **ορατό**. Στο αρχείο **queries.css** και μέσα στην τελευταία **media query** που αναφέρεται σε πλάτη οθόνης κάτω από 944 px (59 em) και μετά τον τελευταίο κανόνα αυτής, προσθέτουμε τον επόμενο ο οποίος για αυτές τις διαστάσεις οθόνης κάνει το κουμπί ορατό.

```
@media (max-width: 59em) {  
    /* 940px/16px=59em */  
  
    >> >> >> >> >> >> >>  
    >> >> >> >> >> >> >>  
    >> >> >> >> >> >> >>  
  
    /* MOBILE NAVIGATION */  
    .btn-mobile-nav {  
        display: block;  
    }  
}
```

Σώζω και βλέπω τις αλλαγές. Αρχίζω να αλλάζω το πλάτος οθόνης. Όσο το πλάτος είναι πάνω από 940px το κουμπί **menu** είναι αόρατο. Μόλις το πλάτος κατέβει κάτω από 940 px το κουμπί **menu** γίνεται ορατό.

Κάτι άλλο που πρέπει να αλλάζει είναι η μπάρα πλοήγησης η οποία τώρα πρέπει να είναι κάθετη και να επικαλύπτει όλη την οθόνη να κάθεται δηλαδή στην κορυφή του περιεχομένου.

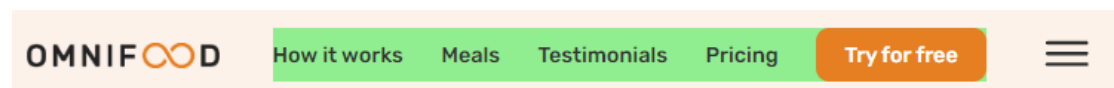
Η μπάρα πλοήγησης βρίσκεται στο εσωτερικό της διπλής ετικέτας `<nav></nav>`. Σ' αυτή την ετικέτα θα πρέπει να εφαρμόσουμε **absolute positioning** και θα την αναγκάσουμε να απλώσει σε όλη την οθόνη.

Κάτω από τον προηγούμενο κανόνα προσθέτω τον επόμενο με τον οποίο δίνω αρχικά ένα χρώμα στο φόντο της μπάρας για να μπορώ να βλέπω τα όριά της.

```
/* MOBILE NAVIGATION */
.btn-mobile-nav {
  display: block;
}

nav {
  background-color: lightgreen;
}
```

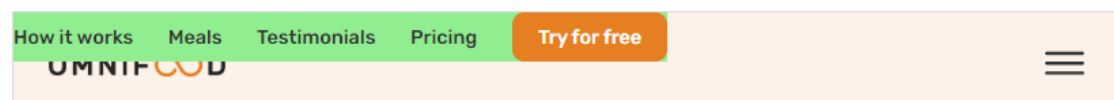
Σώζω και βλέπω τις αλλαγές. Η μπάρα βάφεται πράσινη.



Στη συνέχεια μέσα στον κανόνα της **nav** προσθέτω τις επόμενες ιδιότητες.

```
nav {
  background-color: lightgreen;
  position: absolute;
  top: 0;
  left: 0;
}
```

Ο σκοπός μας είναι να απομακρύνουμε την μπάρα από την κανονική ροή και να την φέρουμε να καθίσει **πάνω από τα περιεχόμενα της σελίδας** ξεκινώντας από το πάνω αριστερό της άκρο.



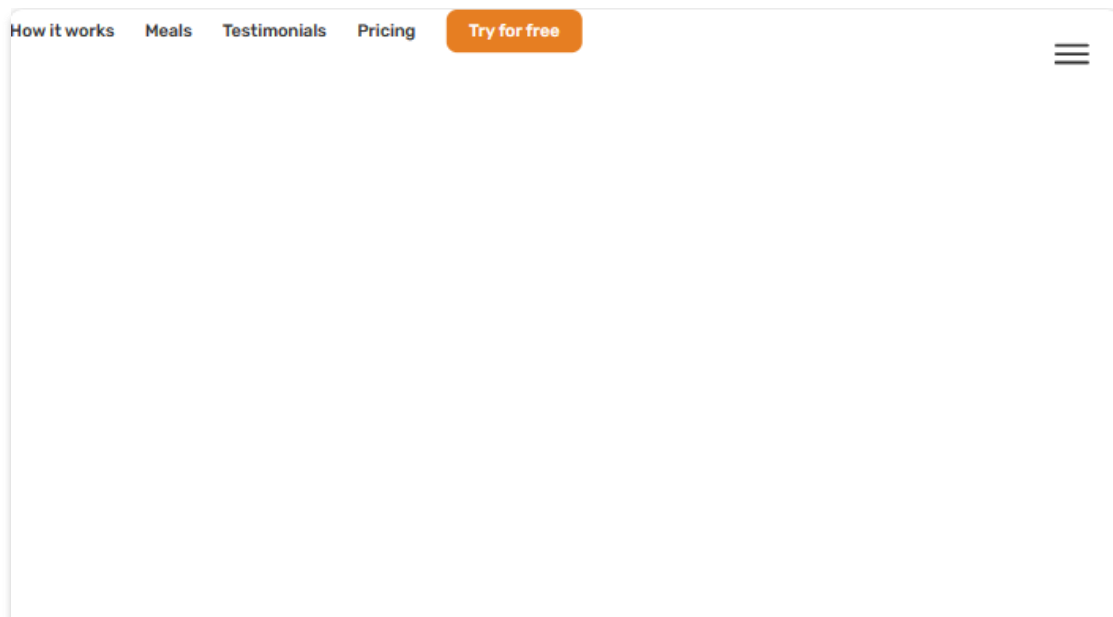
Θυμηθείτε ότι για να επιβάλω absolute positioning σε ένα στοιχείο θα πρέπει να έχω επιβάλλει στο πατρικό του στοιχείο relative positioning.

```
header {  
  position: relative;  
}
```

Θα δώσω στην μπάρα πλοήγησης τη μορφή κατακόρυφου μενού. Επιστρέφω στον προηγούμενο κανόνα και αλλάζω το φόντο της **nav** σε λευκό και μεγαλώνω το κουτί της στα όρια του παραθύρου, δίνοντας στο **πλάτος** της (width) την τιμή **100%** προκειμένου το κουτί της nav να απλώσει σε όλο το πλάτος της οθόνης του χρήστη και στο **ύψος** (height) την τιμή **100vh** προκειμένου να απλώσει σε ύψος στο 100% του ύψους της οθόνης του χρήστη (**viewport height**).

```
nav {  
  background-color: #FFF;  
  position: absolute;  
  top: 0;  
  left: 0;  
  width: 100%;  
  height: 100vh;  
}
```

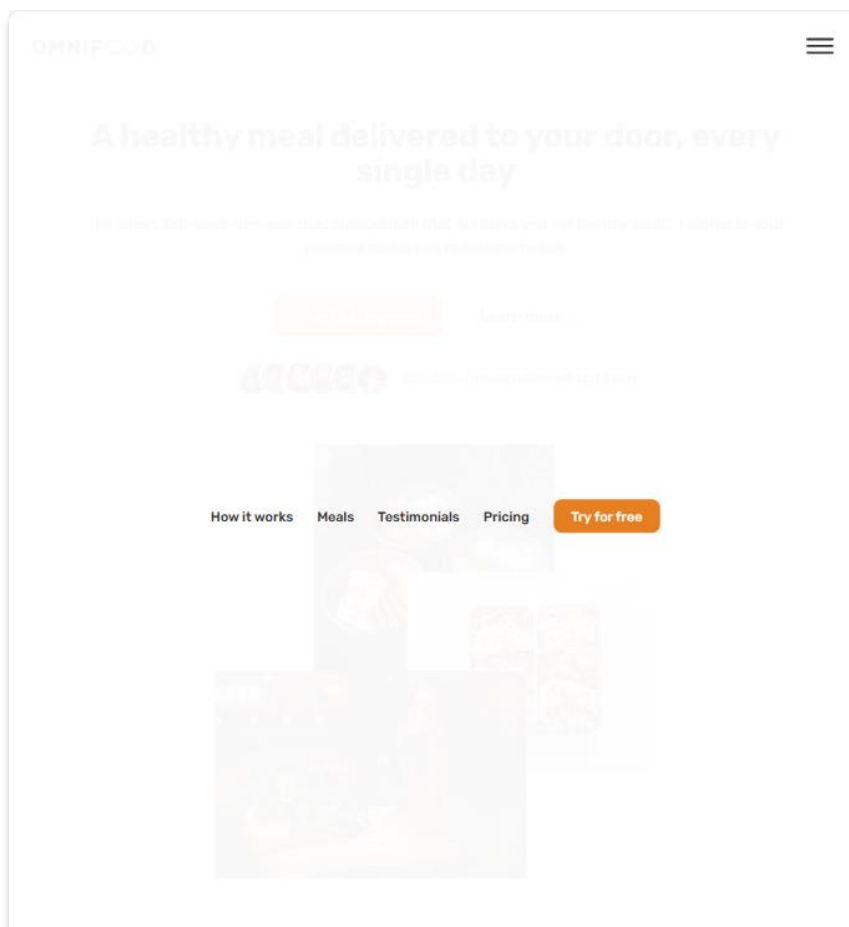
Σώζω και βλέπω τις αλλαγές.



Στον προηγούμενο κανόνα δίνω μια μικρή διαφάνεια στο λευκό υπόβαθρο της nav και στη συνέχεια με τη βοήθεια του flexbox στοιχίζω τη μπάρα πλοήγησης στο κέντρο του κουτιού της nav. Η στοίχιση γίνεται ως προς τον οριζόντιο άξονα και ως προς τον κατακόρυφο.

```
nav {  
  background-color: rgba(255,255,255,0.97);  
  position: absolute;  
  top: 0;  
  left: 0;  
  width: 100%;  
  height: 100vh;  
  
  display: flex;  
  justify-content: center;  
  align-items: center;  
}
```

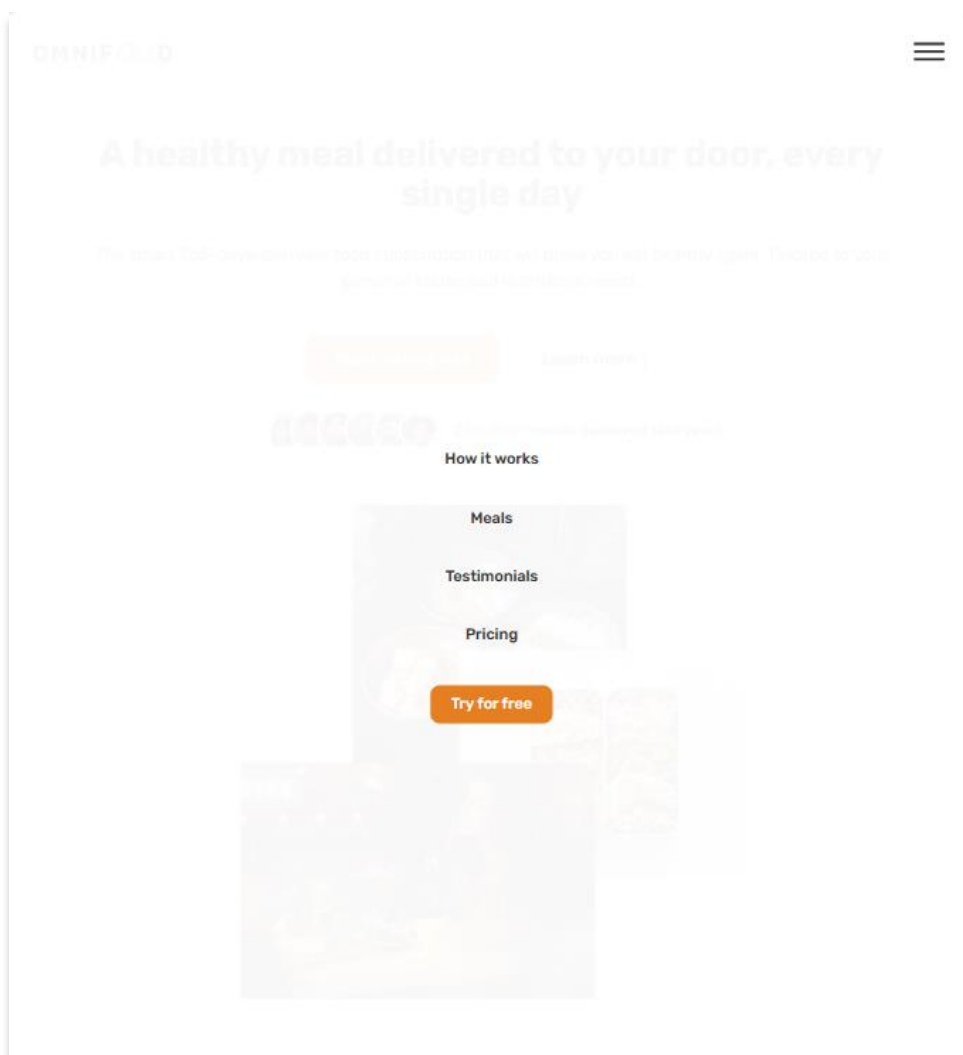
Σώζω και βλέπω τις αλλαγές.



Ακριβώς από κάτω προσθέτω νέο κανόνα που στοχεύει στην **nav-list** και αλλάζει την κατεύθυνση του οριζόντιου άξονα σε κατακόρυφη, στοιχίζει στο κέντρο του άξονα τα στοιχεία και τους δίνει απόσταση 4.8rem.

```
.nav-list {  
  flex-direction: column;  
  justify-content: center;  
  gap: 4.8rem;  
}
```

Σώζω και βλέπω τις αλλαγές.



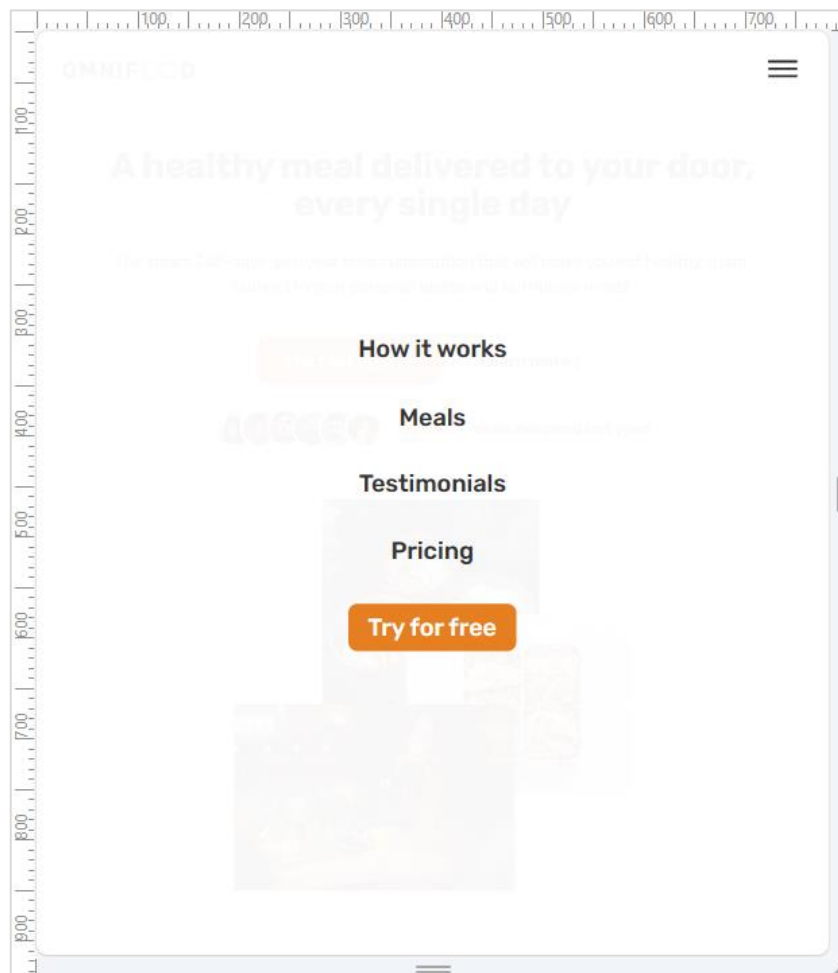
Στη συνέχεια θα αυξήσω το μέγεθος του κειμένου των στοιχείων της μπάρας. Θυμηθείτε ότι τα στοιχεία της μπάρας είναι ετικέτες a και μορφοποιούνται μέσω του επόμενου κανόνα:

```
.btn-menu:link, .btn-menu:visited {
  text-decoration: none;
  font-size: 1.8rem;
  color: #333;
  font-weight: 500;
  display: inline-block;
  transition: all 0.3s;
}
```

Μέσα στο **media query** προσθέτω τον επόμενο κανόνα ο οποίος αυξάνει το μέγεθος των στοιχείων για τα συγκεκριμένα πλάτη οθόνης

```
.btn-menu:link, .btn-menu:visited {
  font-size: 3rem;
}
```

Σώζω και βλέπω τις αλλαγές.



Αρχικά η **nav** θα πρέπει να είναι αόρατη. Στον κανόνα της **nav** προσθέτω την επόμενη ιδιότητα.

```
nav {  
    background-color: rgba(255,255,255,0.97);  
    position: absolute;  
    top: 0;  
    left: 0;  
    width: 100%;  
    height: 100vh;  
  
    display: flex;  
    justify-content: center;  
    align-items: center;  
  
    /*Hide Navigation*/  
    display: none;  
}
```

Η **nav** γίνεται αόρατη. Αυτή η ιδιότητα εξαφανίζει το στοιχείο nav από τη ροή, όμως δεν επιτρέπει την χρήση **animation** – κίνησης.

Αντί αυτής θα χρησιμοποιήσουμε την ιδιότητα **opacity** με τιμή **0** προκειμένου να αποκρύψουμε την **nav**.

```
nav {  
    background-color: rgba(255,255,255,0.97);  
    position: absolute;  
    top: 0;  
    left: 0;  
    width: 100%;  
    height: 100vh;  
  
    display: flex;  
    justify-content: center;  
    align-items: center;  
  
    /*Hide Navigation*/  
    opacity: 0;  
}
```

Μηχανισμός Εμφάνισης / Απόκρυψης του Μενού

Ο μηχανισμός θα βασίζεται σε μία νέα κλάση που θα προσθέσουμε στην ετικέτα **header** με το όνομα **nav-open** στο αρχείο **index.html**.

```
<header class="nav-open">
.....
.....
</header>
```

Την κλάση αυτή θα προσθέτουμε ή θα απομακρύνουμε από την ετικέτα αρχής **<header>** με τη βοήθεια κώδικα **JavaScript**, καθώς ο χρήστης θα πατά διαδοχικά στο κουμπί της πλοήγησης. Όταν η κλάση είναι παρούσα η πλοήγηση θα είναι ορατή. Στο αρχείο **queries.css** προσθέτω τον επόμενο κανόνα κάτω από τον κανόνα που στοχεύει στην **nav**.

```
.nav-open nav {
    opacity: 1;
}
```

Επίσης στον κανόνα της **nav** προσθέτω την ιδιότητα **transition** προκειμένου η εμφάνιση ή η απόκρυψη της πλοήγησης να γίνεται σταδιακά.

```
nav {
    background-color: rgba(255,255,255,0.97);
    position: absolute;
    top: 0;
    left: 0;
    width: 100%;
    height: 100vh;

    display: flex;
    justify-content: center;
    align-items: center;
    transition: all 0.5s;

    /*Hide Navigation*/
    opacity: 0;
}
```

Για να αποκρύψουμε όμως πλήρως την περιοχή πλοήγησης δεν αρκεί να την κάνουμε αόρατη, γιατί παρότι η περιοχή δεν φαίνεται πλέον, οι σύνδεσμοι που υπάρχουν μέσα σ' αυτήν συνεχίζουν να λειτουργούν. Έτσι αν κάποιος πατήσει μέσα στην περιοχή με το ποντίκι υπάρχει η πιθανότητα να πατήσει πάνω σε κάποιον από τους συνδέσμους που υπάρχουν εκεί απλά δεν φαίνονται.

Έτσι μια δεύτερη ενέργεια που θα χρειαστεί να κάνουμε θα είναι η απενεργοποίηση του ποντικιού και του πληκτρολογίου για την περιοχή πλοήγησης. Αυτό γίνεται με την προσθήκη της ιδιότητας **pointer-events** με τιμή **none** στον κανόνα της **nav**.

Και μια τρίτη ενέργεια που απαιτείται είναι η απόκρυψη της περιοχής πλοήγησης από τους αναγνώστες οθόνης. Αυτό επιτυγχάνεται με τη χρήση της ιδιότητας **visibility** με τιμή **hidden**.

```
nav {  
    background-color: rgba(255,255,255,0.97);  
    position: absolute;  
    top: 0;  
    left: 0;  
    width: 100%;  
    height: 100vh;  
  
    display: flex;  
    justify-content: center;  
    align-items: center;  
    transition: all 0.5s;  
  
    /*Hide Navigation*/  
    opacity: 0;  
    pointer-events: none;  
    visibility: hidden;  
}
```

ΠΡΟΣΟΧΗ! Όταν θα εμφανίζουμε την πλοήγηση ξανά θα πρέπει να τα επαναφέρουμε όλα.

```
.nav-open nav {  
    opacity: 1;  
    pointer-events: auto;  
    visibility: visible;  
}
```

Επίσης όταν η πλοήγηση είναι ορατή το κουμπί menu πρέπει να εξαφανίζεται και στη θέση του να εμφανίζεται το close. Η πλοήγηση είναι ορατή όταν η ετικέτα αρχής **header** συνοδεύεται από την κλάση **nav-open**. Γι' αυτή την περίπτωση προσθέτω τους επόμενους 2 κανόνες ο μεν πρώτος στοχεύει στο κουμπί close και το εμφανίζει, ο δεύτερος αποκρύπτει το κουμπί μενού.

```
.nav-open .icon-mobile-nav[name="close-outline"] {  
  display: block;  
}  
  
.nav-open .icon-mobile-nav[name="menu-outline"] {  
  display: none;  
}
```

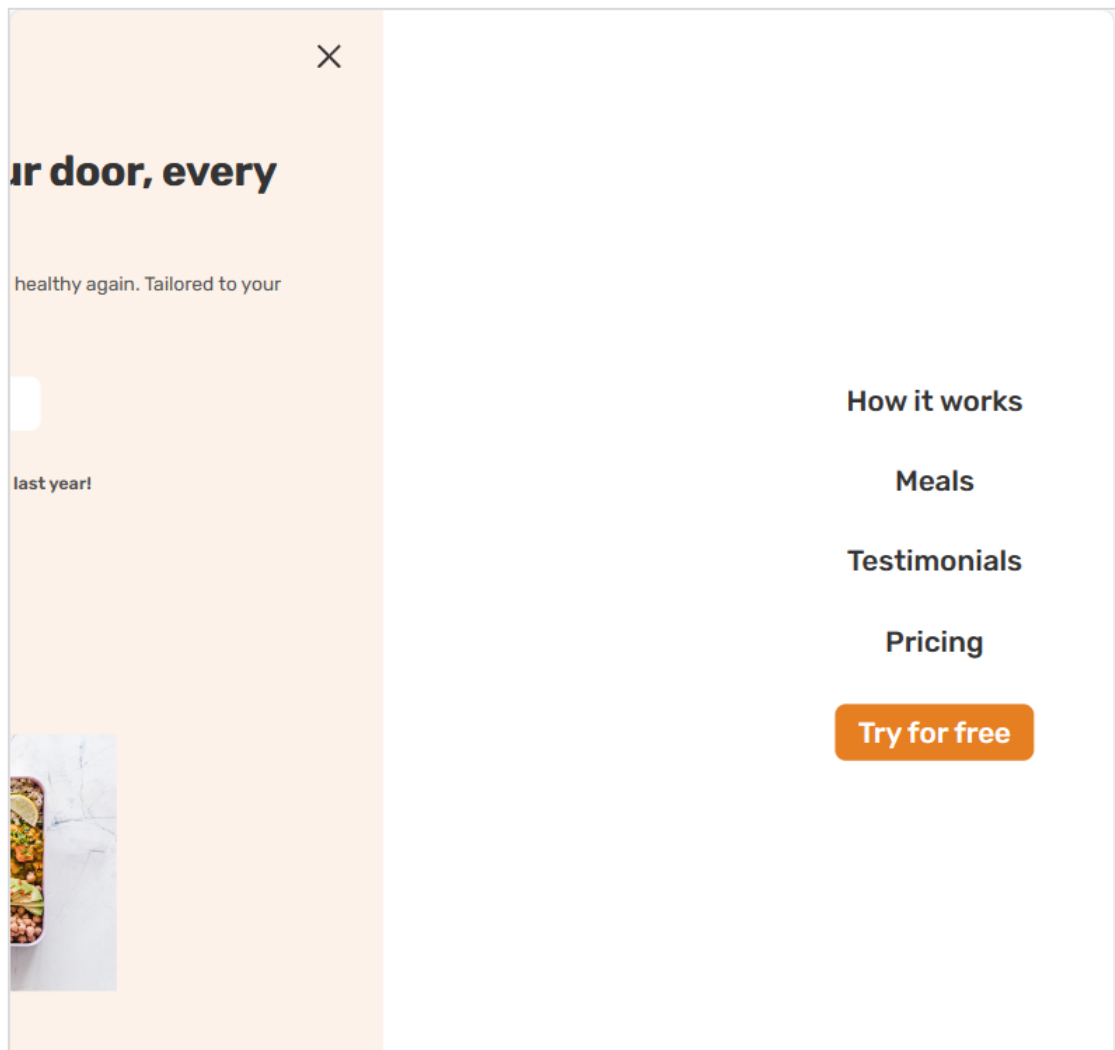
Η απόκρυψη/εμφάνιση αυτών των κουμπιών μπορεί να γίνει με την βοήθεια της ιδιότητας **display** μιας και στα συγκεκριμένα κουμπιά δεν μας ενδιαφέρει η αλλαγή να συνοδεύεται από κάποια κίνηση (animation).

Προσθήκη Κίνησης (Animation)

Αρχικά μετακινώ όλο το κουτί της πλοήγησης έξω από το παράθυρο σέρνοντάς το στον άξονα X προσθέτοντας την ιδιότητα **transform** με τιμή **translateX(100%)** στον κανόνα **nav**.

```
nav {  
  background-color: rgba(255,255,255,0.97);  
  position: absolute;  
  top: 0;  
  left: 0;  
  width: 100%;  
  height: 100vh;  
  transform: translateX(100%);  
  
  display: flex;  
  justify-content: center;  
  align-items: center;  
  transition: all 0.5s;  
  
  /*Hide Navigation*/  
  opacity: 0;  
}
```

Το πρόβλημα όμως είναι ότι αν σύρω προς τα αριστερά θα δω το κουτί της πλοήγησης.



Για να το αποκρύψω προσθέτω την ιδιότητα **overflow-x** με τιμή **hidden** στους κανόνες της **body** και της **html** στο αρχείο **style.css**.

```
html {  
  font-size: 62.5%;  
  overflow-x: hidden;  
}  
  
body {  
  font-family: "Rubik", sans-serif;  
  color: #555;  
  overflow-x: hidden;  
}
```

Αν αντί της `overflow-x` χρησιμοποιήσω την `overflow` δεν θα μπορώ να ρολλάρω προς τα κάτω στον άξονα Y και δεν θα έχω πρόσβαση στο υπόλοιπο περιεχόμενο της σελίδας.

Όταν είναι παρούσα η κλάση **nav-open** στην ετικέτα αρχής **header** το κουτί της πλοήγησης πρέπει να εμφανίζεται ξανά. Στον επόμενο κανόνα το επαναφέρω στην αρχική του θέση με την ιδιότητα **transform** και τιμή **translate(0)**.

```
.nav-open nav {  
  opacity: 1;  
  pointer-events: auto;  
  visibility: visible;  
  transform: translateX(0);  
}
```

Ακολουθεί ο κώδικας του media query όπως έχει διαμορφωθεί ως τώρα

```
@media (max-width: 59em) {  
  /* 940px/16px=59em */  
  
  html {  
    font-size: 50%;  
  }  
  
  .hero {  
    grid-template-columns: 1fr;  
    padding: 0 8rem;  
    gap: 6.4rem;  
  }  
  
  .hero-img {  
    width: 60%;  
  }  
  
  .hero-text-box, .hero-img-box {  
    text-align: center;  
  }  
  
  .delivered-meals {  
    justify-content: center;  
    margin-top: 3.2rem;  
  }  
  
  .logos img {  
    height: 2.4rem;  
  }  
  
  .step-number {  
    font-size: 7.4rem;  
  }  
  
  .meal-content {  
    padding: 2.4rem 3.2rem 3.2rem 3.2rem;  
  }  
}
```

```

.section-testimonials {
    grid-template-columns: 1fr;
}

.gallery {
    grid-template-columns: repeat(6, 1fr);
}

.cta {
    grid-template-columns: 3fr 2fr;
}

.cta-form {
    grid-template-columns: 1fr;
}

.btn-form {
    margin-top: 1.2rem;
}

.gallery-item:nth-child(11), .gallery-item:nth-child(12) {
    display: inline;
}

/* MOBILE NAVIGATION */

.btn-mobile-nav {
    display: block;
}

header {
    position: relative;
}

nav {
    background-color: rgba(255,255,255,0.97);
    position: absolute;
    top: 0;
    left: 0;
    width: 100%;

```

```
height: 100vh;
transform: translateX(100%);

display: flex;
justify-content: center;
align-items: center;
transition: all 0.5s ease-in;

/*Hide Navigation*/
opacity: 0;
}

.nav-open nav {
  opacity: 1;
  pointer-events: auto;
  visibility: visible;
  transform: translateX(0);
}

.nav-open .icon-mobile-nav[name="close-outline"] {
  display: block;
}

.nav-open .icon-mobile-nav[name="menu-outline"] {
  display: none;
}

.nav-list {
  flex-direction: column;
  justify-content: center;
  gap: 4.8rem;
}

.btn-menu:link, .btn-menu:visited {
  font-size: 3rem;
}
}
```