

# Дадлагын ажлын тайлан

МТ, 20В1NUM1201, Г.Зандан-Очир

# Агуулга

- 01 БАЙГУУЛЛАГЫН ТАНИЛЦУУЛГА
- 02 ДАДЛАГЫН АЖЛЫН ТӨЛӨВЛӨГӨӨ
- 03 СИСТЕМИЙН ШААРДЛАГА, СУДАЛГАА
- 04 АШИГЛАСАН ТЕХНОЛОГИ
- 05 ХЭРЭГЖҮҮЛЭЛТ



01

# БАЙГУУЛЛАГЫН ТАНИЛЦУУЛГА

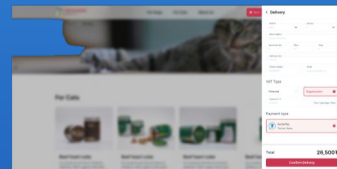
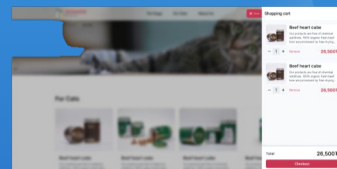
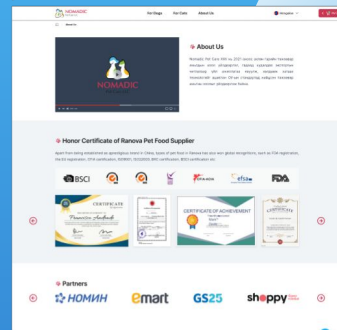
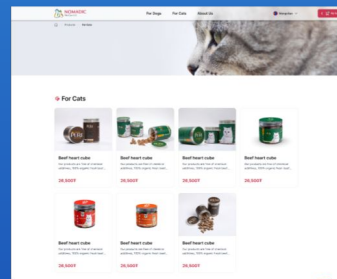
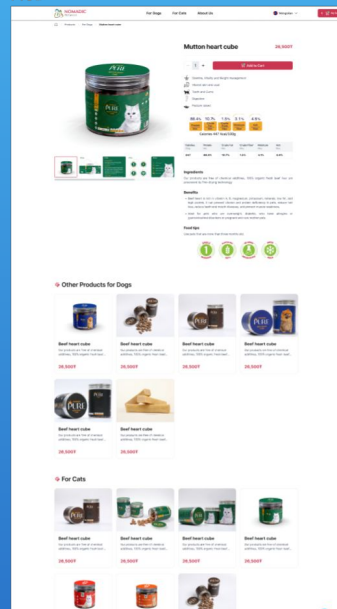
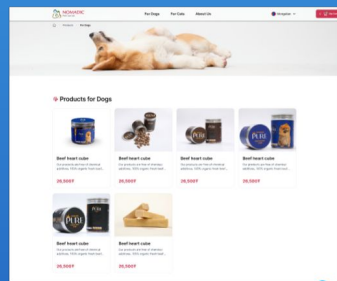
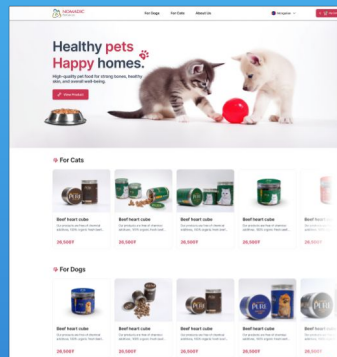
# TECH PARTNERS LLC

The logo for Tech Partners is centered within a large white circle. It features the word "TECH" in a large, bold, blue sans-serif font. Below "TECH", the word "PARTNERS" is written in a smaller, blue, spaced-out sans-serif font.

**TECH**  
P A R T N E R S

Тек Парнтерс ХХК нь 2020 оноос эхлэн Teso группын хөрөнгө оруулалт аван **FIBO CLOUD** компаниас салбарлан клауд консалтинг, систем хөгжүүлэлтээр дагнан үйл ажиллагаа явуулж эхэлсэн.

Ахлах хөгжүүлэгч **Д. Бат-Оргилын** удирдлага дор веб фронт энд хөгжүүлэлт, эцсийн бүтээгдэхүүн хөгжүүлэн гаргах талаар суралцаж үүний дагуу **"Nomadic Pet Care LLC"** компанийн бараа бүтээгдэхүүн борлуулах Ecommerce системийн фронт-энд хөгжүүлэлтийг хийж гүйцэтгэлээ.





02

# ДАДЛАГЫН АЖЛЫН ТӨЛӨВЛӨГӨӨ

<u>Гүйцэтгэх ажил</u>	Хугацаа
<u>Технологиудыг судлах</u>	06/12
<u>Дизайнтай танилцаж, төсөлөө эхлүүлэх</u>	06/13
<u>Гурван хэл дээр ажилдаг болгох</u>	06/14 - 06/15
<u>Нүүр хуудас, Бидний тухай хуудас хийх</u>	06/18 - 06/20
Динамик бүтээгдхүүний дэлгэрэгүй, ангилалын дэлгэрэнгүй хуудас хийх	06/21 - 06/23
Бараа сагслах логик хийх	06/25
Хүргэлтийн мэдээлэл авах форм	06/26
Төлбөр төлөх логик	06/27
Тестенд өгөх, тестийн засвар, сайжруулалт хийх	06/28 - 06/30



03

# **СИСТЕМИЙН ШААРДЛАГА, СУДАЛГАА**



# ИЖИЛ СИСТЕМИЙН СУДАЛГАА



**chewy.com**

React, Next.js, Material UI, i18next



**shoppymn**

React, хэрэглэгчийн хүлээлт



Бараа бүтээгдэхүүн  
худалдан авах  
хэрэглэгчид

## Функционал шаардлага

- Хэрэглэгч хэлээ солиж болдог байх
- Хэрэглэгч барааг төрлөөр жагсаалт хэлбэрээр хардаг байх
- Жагсаалтнаас бүтээгдхүүний нэр, богино тайлбар, үнэ харж болдог байх
- Бүтээгдэхүүний дэлгэрэнгүй харж болдог байх
- Хэрэглэгч сагсанд бараа хийх, хасах, тоо хэмжээг өөрчилж болдог байх
- Сагсанд хийсэн бараа дараа ороход байдаг байх
- Хүргэлтийн мэдээлэлээ хийж болдог байх
- Төлбөр төлөлт хийж болдог байх



Бараа бүтээгдэхүүн  
худалдан авах  
хэрэглэгчид

## Функционал бус шаардлага

- Сайт ямар ч төхөөрөмж дээр ажилдаг responsive байх
- Харилцан үйлдэл сайтай буюу хэрэглэгчид яг юу болж байгааг мэдээлэлдэг
- Ашиглахад ойлгомжтой байх
- Хүртээмжтэй байх



04

**АШИГЛАСАН ТЕХНОЛОГИ**



**Git/Gitlab**



**React**



**Next.js**



**TypeScript**



**Mantine UI**



**i18next /  
next-i18next**



05

**ХЭРЭГЖҮҮЛЭЛТ**



# Төсөл эхлүүлэх

Төслөө эхлүүлэхдээ хамгийн эхэнд хөгжүүлэлтийн орчноо бэлдэх зорилгоор **Next.js**-ийн **create-next-app** командаар шинэ төсөл эхлүүлж **Mantine UI**, **i18next**, **next-i18next** зэрэг сангуудыг нэмэн public фолдэр дотроо орчуулгын файлуудыг нэмэв.

Дараагаар системийн стайлыг хариуцах **Mantine UI** сангийн **Provider** дотор үндсэн өнгө, фонт зэргийг тохируулан, **next-i18next** сангийн **Provider**-ийг тохируулж, тохиргооны файлыг төсөлд нэмэв.

```
/** @type {import('next-i18next').UserConfig} */  
  
module.exports = {  
  i18n: {  
    defaultLocale: "en",  
    locales: ["en", "mn", "cn"],  
  },  
};
```

```
import { useRequest } from "ahooks";  
import { Options, Service } from  
"ahooks/lib/useRequest/src/types";  
import { i18n } from "next-i18next";  
  
function useApi<TData, TParams extends any[]>(  
  service: Service<TData, TParams>,  
  options?: Options<TData, TParams>  
) {  
  const api = useRequest(service, {  
    ...options,  
    refreshDeps: [i18n?.language, ...(options?.refreshDeps ||  
    [])];  
  });  
  return api;  
}  
  
export default useApi;
```

# Нийтлэг component

Вебсайтны бүх эсвэл ихэнх хуудасанд хэрэглэгдэх component-ууд

```
const categoryListAPI = useApi(() => reference.getCategoryList({}),  
{ onError: (error) => {  
  Error(error.message);  
}},  
{});
```

```
const menus = categoryListAPI?.data  
?.sort((a, b) => {  
  return a.rank - b.rank;  
})  
?.map((e) => {  
  return (  
    <Link  
      href={{  
        pathname:  
          `/category/${e.id}`,  
        onClick={() => closeDrawer()}  
      },  
      className={classes.link}  
    > {e.name}  
    </Link>  
  );  
});
```



[For cats](#) [For dogs](#) [About Us](#)



English ▾



My Cart



## Snack for Dog - 60gram

100% organic fresh beef heart are processed by freeze-drying technology

21,000₹



# Статик хуудаснууд

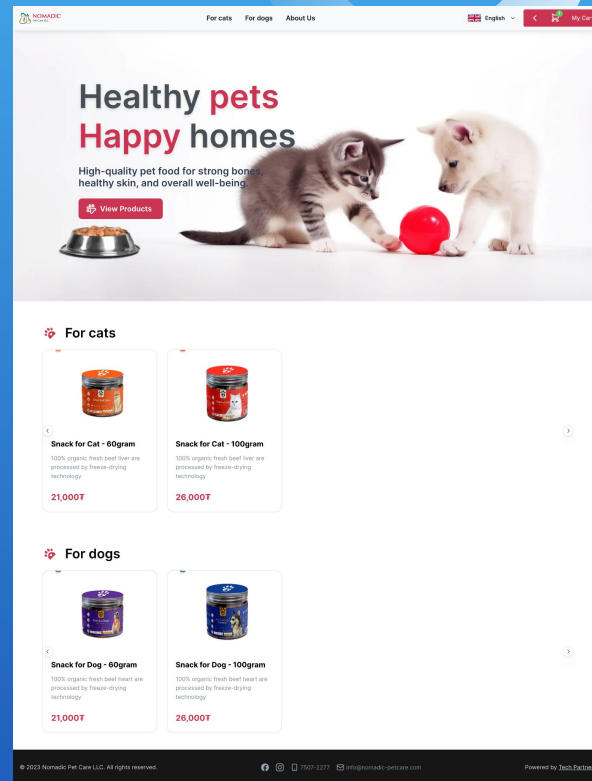
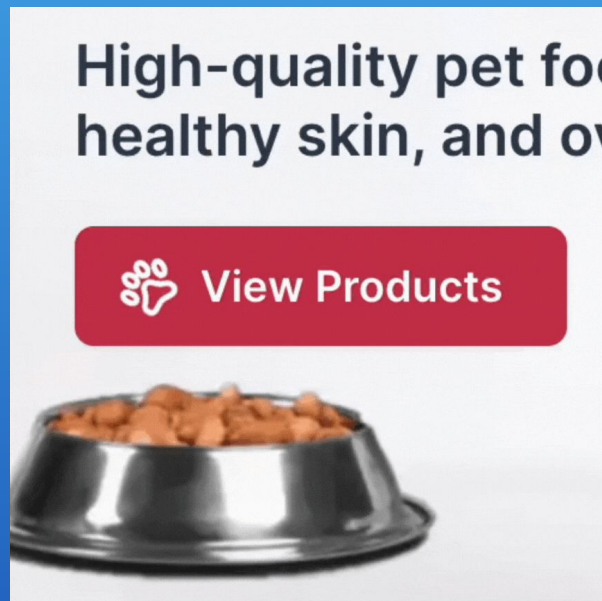
```
const jiggle = keyframes({
  '0%': { rotate: '0deg' },
  '5%': { rotate: '-10deg' },
  '10%': { rotate: '10deg' },
  '15%': { rotate: '0deg' },
  '20%': { rotate: '10deg' },
  '25%': { rotate: '-10deg' },
  '30%': { rotate: '0deg' },
  '35%': { rotate: '-10deg' },
  '40%': { rotate: '10deg' },
  '45%': { rotate: '0deg' },
  '50%': { rotate: '-10deg' },
  '60%': { rotate: '0deg' },
  '70%': { rotate: '0deg' },
  '80%': { rotate: '0deg' },
  '90%': { rotate: '0deg' },
  '100%': { rotate: '0deg' },
});

...

icon: {
  transition: "transform 100ms ease",
  rotate: "-45deg",

  "&:hover": {
    transform: "scale(1.2)",
    animation: `${jiggle} 2s ease-in-out`,
  },
},

...
```

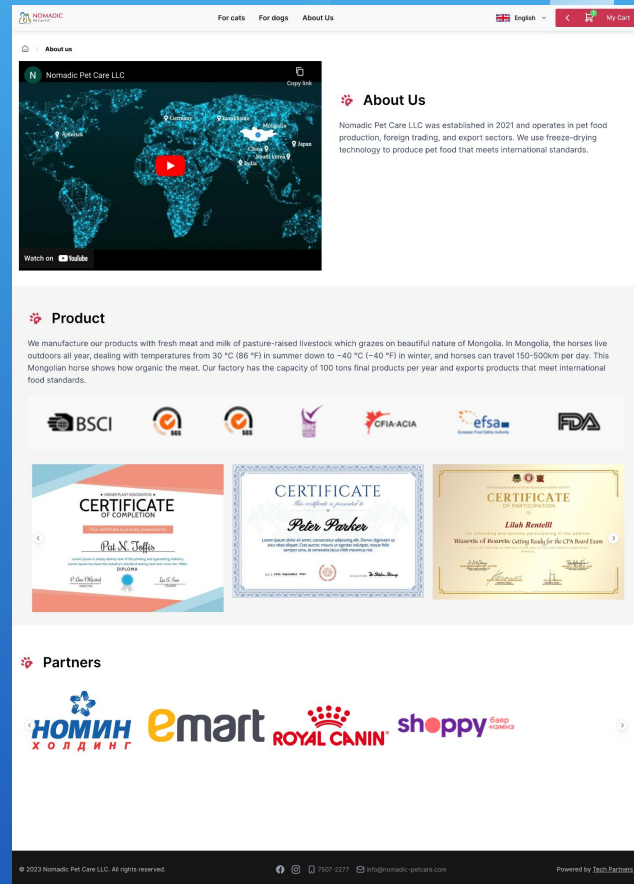


# Статик хуудаснууд

```
const YoutubeEmbed = ({ url }: { url: string }) => {
  const id = extractVideoId(url);
  return (
    <div className="video-responsive">
      <iframe
        width="100%"
        height="480"
        src={`https://www.youtube.com/embed/${id}`}
        frameBorder="0"
        allow="accelerometer; autoplay; clipboard-write; encrypted-media; gyroscope; picture-in-
        picture"
        allowFullScreen
        title="Embedded youtube"
      />
    </div>
  );
};

function extractVideoId(url: string): string | null {
  const regex = /(?:\|\/|3D|v=|vi=)([0-9a-zA-Z_-]{11})(?:[?&]|$)/;
  const match = url?.match(regex);
  return match ? match[1] : null;
}

export default YoutubeEmbed;
```



# Динамик хуудаснууд

Өмнө дурдсанчлан **Next.js** санг ашиглах том шалтгаан нь Search Engine Optimization билээ. Хуудсаа хэрэглэгчийн интернэт хөтөч дээр биш сервер дээр дүрслэл(render) хийснээр хайлтын системийн бот crawl хийж хайлт сайжрах юм.

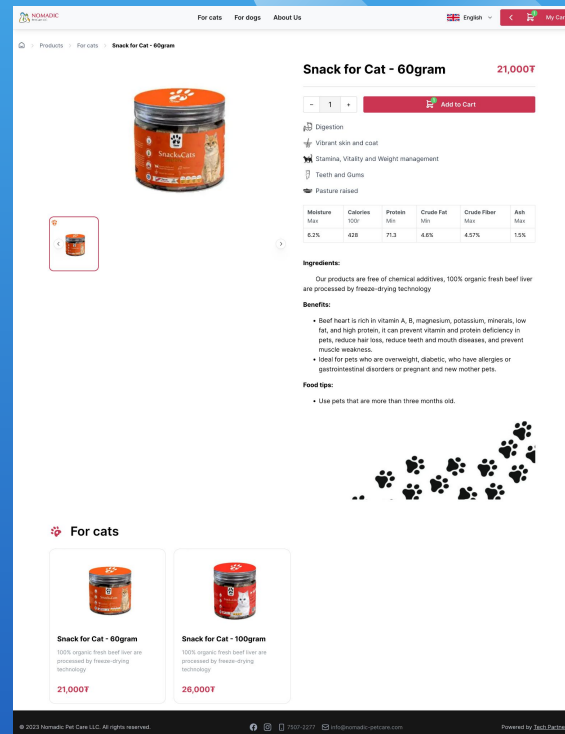
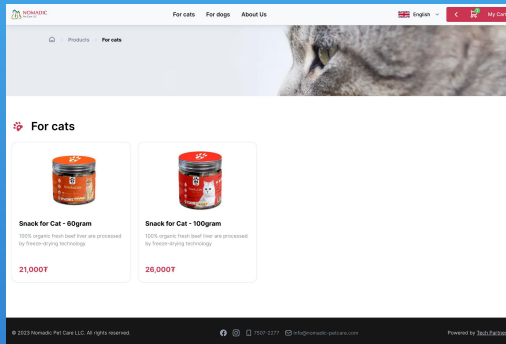
Дараах код нь бүтэгдэхүүний мэдээллийг болон орчуулгыг сервер дээрээ дуудан дүрслэл хийнэ. Бүтэгдэхүүний id-ийг дуудаж буй хуудаснаасаа буюу path-наас авах ба **Next.js**-ийн **GetServerSideProps** функц дээр сервер дээр хийгдэх бүх дата дуудалт мөн **next-i18next** сангийн **serverSideTranslations** функцээр серверийн орчуулгаа хийв.

<https://nomadic-petcare.com/category/2>

```
...
```

```
const CategoryDetailPage: FC<{ data: Category }> = ({ data }) => {  
  const { t } = useTranslation();  
  return (  
    <Page title={data?.name || t("Nomadic Pet Care LLC")}>  
      <CategoryDetail detail={data} />  
    </Page>  
  );  
};
```

```
export const getServerSideProps: GetServerSideProps = async ({  
  query,  
  locale,  
}) => {  
  const { id } = query as unknown as QueryProps;  
  try {  
    let all = await reference.getCategoryList({  
      locale: locale || "en",  
    });  
    let data = all.find((item) => item.id === Number(id));  
    if (!data) throw new Error("Category not found");  
    return {  
      props: {  
        data: data,  
        ...(await serverSideTranslations(locale || "en",  
["common"])),  
      };  
    } catch (err) {  
      console.log("error", err);  
      return {  
        notFound: true,  
      };  
    }  
  }  
};  
  
export default CategoryDetailPage;
```



# useCart хүүк

Хэрэглэгч сагсанд бараа хийх, хасах, тоо хэмжээг өөрчлөх зэрэг сагстай холбоотой бүх үйлдлийг нэг аргаар, нэг сувгаар зохиулахын тулд дараах хүүкийг бичив.

Мөн хэрэглэгчийн сагсанд нь байгаа барааны мэдээллийг интернэт хөтчийн дотоод санах ойд хадгалснаар хуудас хооронд өгөгдөл дамжуулах, хэрэглэгч дараа хандахад бараанууд байх зэрэг хоёр асуудлыг шийдэж байгаа юм.

```

export interface Item {
  note?: string;
  product_id: number;
  qty: number;
  product: Product;
}

const useCart = () => {
  const { data: productList } = useApi(() => reference.getProductList({}));
  const [cartItems, setCartItems] = useLocalStorage<Item[]>({
    key: MyCartKey.items,
    defaultValue: [],
  });

  const onQuantityChange = (id: number, qty: number) => {
    const product = productList?.find((i) => i.id === id);
    if (!product) {
      notifications.show({
        title: "Product not found",
        message: "Please refresh the page",
      });
      return;
    }
    const item = cartItems.find((i) => i.product_id === id);
    if (item) {
      qty === 0
        ? setCartItems(cartItems.filter((i) => i.product_id !== id))
        : setCartItems(
            cartItems.map((i) => (i.product_id === id ? { ...i, qty } : i))
          );
    }
    if (!item) {
      setCartItems([...cartItems, { product_id: id, qty, product, note: ""
    }]);}
  };

  const getCartItem = (id: number) => {
    const product = productList?.find((i) => i.id === id);
    if (!product) {
      notifications.show({
        title: "Product not found",
        message: "Please refresh the page",
      });
      return;
    }
    const item = cartItems.find((i) => i.product_id === id);
    if (item) {
      return { ...item, product };
    }
    return undefined;
  };
};

```

```

const addToCart = (id: number, qty: number) => {
  const product = productList?.find((i) => i.id === id);
  if (!product) {
    notifications.show({
      title: "Product not found",
      message: "Please refresh the page",
    });
    return;
  }
  const item = cartItems.find((i) => i.product_id === id);
  if (item) {
    setCartItems(
      cartItems.map((i) =>
        i.product_id === id ? { ...i, qty: i.qty + qty } : i
      )
    );
  }
  if (!item) {
    setCartItems([...cartItems, { product_id: id, qty, product, note: ""
  }]);}
};

const clearCart = () => {
  setCartItems([]);
};

const getItemsCount = () => {
  return cartItems.reduce((acc, item) => acc + item.qty, 0);
};

return {
  cartItems,
  onQuantityChange,
  getCartItem,
  addToCart,
  clearCart,
  getItemsCount,
};

export default useCart;

```

```
const { getItemCount, onQuantityChange } =  
  useCart();  
...  
  
<Button  
  disabled={count === 0}  
  ...  
  onClick={() => {  
    onQuantityChange(detail.id, count);  
  }}  
>  
  <CartWithNumber number={getItemCount()} />  
  {t("Add to Cart")}  
</Button>
```



Add to Cart

```

const renderFlyingBones = () => {
  if (showFlyingBones) {
    return (
      <div className={classes.flyingBonesContainer}>
        {Array.from({ length: 5 }, (_, index) => (
          <IconBone
            key={index}
            style={{
              position: 'absolute',
              top: 0,
              left: 0,
              animation: `fly-bone-${index} 2s linear`,
              color: '#FBB71A',
            }}
          />
        ))}
      </div>
    );
  }
};

<style>
  {Array.from(
    { length: 5 },
    (_, index) =>
      @keyframes fly-bone-${index} {
        0% {
          transform: translate(0, 0);
        }
        100% {
          transform: translate(${getRandomInt(-100, 100)}px,
            ${getRandomInt(-100, 100)}px);
        }
      }
    )}
</style>

```

at - 100gram



Add to C



```

const { cartItems, onQuantityChange, getItemsCount } =
useCart();
const totalPrice = cartItems.reduce(
  (acc, item) => acc + item.product?.total_price * item.qty,
  0
);

const cartItemsList = cartItems?.map((item, index) => {
  return (
    <>
    <CartItem
      key={item.product.id}
      item={item.product}
      count={item.qty}
      setCustomCount={onQuantityChange}
      remove={() => onQuantityChange(item.product.id, 0)}
    />
    <Divider key={index} color={theme?.colors.gray[3]} />
    </>
  );
});

```

## Миний сагс



### Snack for Cat - 60gram

100% organic fresh beef liver are processed by freeze-drying technology



Устгах

21,000₮



### Нохойнд зориулсан амттан - 60гр

100% үхрийн зүрхийг хөлдөөж хатаан боловсруулсан нэг орцтой бүтээгдэхүүн



Устгах

21,000₮

Нийт

Activate Windows

Go to Settings to activate Windows

42,000₮

Худалдан авах

```

const form = useForm<OrderInput>({
  validate: {
    phone: (value) => {
      if (!value) {
        return t("Phone is required");
      }
      if (value.length < 8) {
        return t("Phone is invalid");
      }
      return null;
    },
    email: (value) => {
      if (!value) {
        return t("Email is required");
      }
      if (!value.match(/[a-zA-Z0-9-_.+!@][a-zA-Z0-9-_.+!@]/)) {
        return t("Email is invalid");
      }
      return null;
    },
    district: (value) => {
      if (!value) {
        return t("District is required");
      }
      return null;
    },
    khoroo: (value) => {
      if (!value) {
        return t("Khoroo is required");
      }
      return null;
    },
    org_req_no: (value) => {
      if (!value) {
        return t("Organization ID is required");
      }
      if (!value || value.length < 7) {
        return t("Organization ID is required");
      }
      if (!value.orgName) {
        return t("Can't find organization");
      }
      if (!value.orgName || value.orgName === "Not Found") {
        return t("Can't find organization");
      }
      return null;
    },
    address_note: (value) => {
      if (!value) {
        return t("Address note is required");
      }
      return null;
    },
    micro_district: (value) => {
      if (!value) {
        return t("Micro district is required");
      }
      return null;
    },
  },
});

```

```

<form onSubmit={form.handleSubmit(values)}>
  <Drawer.Body>
    ...
    <Stack>
      ...
      <Box>
        <Grid columns={6}>
          <Grid.Col span={3}>
            <Select
              size="md"
              label={
                <Text className={classes.selectLabel}>{t("District")}</Text>
              }
              placeholder="Pick one"
              data={
                districts?.map((item) => ({
                  value: item.id.toString(),
                  label: item.region_name,
                }) || [])
              }
              style={{
                position: "relative",
              }}
              styles={{
                input: {
                  backgroundColor: theme.colors.gray[1],
                  borderRadius: "8px",
                  fontSize: "14px",
                },
                item: {
                  fontSize: "14px",
                },
              }}
              {...form.getInputProps("district")}
            </Grid.Col>
          </Grid>
        </Box>
      </Stack>
      <Stack p="md" spacing="md" className={classes.checkoutContainer}>
        <Flex justify="space-between">
          <Text>{t("Total")}</Text>
          <Text className={classes.moneyText}>{moneyFormat(totalPrice)}</Text>
        </Flex>
        <Button
          className={classes.checkoutButton}
          type="submit"
          loading={creating}
        >
          <Text>{t("Confirm Delivery")}</Text>
        </Button>
      </Stack>
    </Drawer.Body>
  </form>

```

## < Хүргэлт

Дүүрэг

Хороо

Pick one

Pick one

Micro district

3, 4-р хороолол

Байр

Давхар

Тоот

6

14

6

Дэлгэрэнгүй хаяг

Address

Дугаар

Имэйл

99999999

Enter email

## VAT Type

Хувь хүн

Байгууллага

## Payment Type



Social Pay

Голомт банк

Нийт

21,000₮

Баталгаажуулах

# Төлбөр төлөлт

Хүргэлтийн форм бөглөж дууссаны дараа back-end луу төлбөр төлөх хүсэлт явуулах ба back-end дээр нэхэмжлэл үүсч хүсэлтэнд 200 буюу амжилттай гэсэн хариу ирвэл Social Pay үйлчилгээний төлбөр төлөх хуудас руу хэрэглэгчийг чиглүүлнэ.

```
...  
  
const { run: create, loading: creating } = useApi(order.create, {  
  manual: true,  
  onSuccess: (data) => {  
    window.location.replace(  
  
`https://commerce.golomtbank.com/socialpay/mn/${data.bank_invoice_id}`  
    ),  
    onError: (error) => {  
      Error(error.message);  
    },  
  
const onSubmit = (values: any) => {  
  create({  
    ...values,  
    district: districts?.find(  
      (district) => district.id === Number(values.district)  
    )?.region_name,  
    khoroo: khoroo?.find((khoroo) => khoroo.id === Number(values.khoroo))  
      ?.region_name,  
    items: cartItems,  
    is_org: isOrg,  
    org_name: orgName,  
    phone: values.phone.toString(),  
  });  
};  
  
...
```

# Төлбөр төлөлт

Хэрэглэгч төлбөрөө амжилттай төлбөл эсвэл амжилтгүй болвол уг сервис буцааж манай order/detail хуудас руу нэхэмжлэлийн дугаарыг араас залгуулан хэрэглэгчийг илгээх ба тухайн хуудас дээр төлбөр төлсөн үгүйг шалгаж амжилттай тохиолдолд нүүр хуудас руу чиглүүлнэ.

```
const router = useRouter();
const { invoice } = router.query as { invoice: string };
const { clearCart } = useCart();
const {} = useApi(
  () => {
    order.check({
      order_uuid: invoice,
    }),
    {
      onSuccess: (data) => {
        if (data.is_paid) {
          clearCart();
        }
        router.push({
          pathname: "/",
          query: {
            paid: data.is_paid,
          },
        });
      },
      ready: !!invoice,
    }
  );
```

For cats
For dogs
About Us
English
My Cart

# Healthy pets Happy homes

High-quality pet food for strong bones, healthy skin, and overall well-being.

[View Products](#)

## For cats

**Snack for Cat - 60gram**

100% organic fresh beef liver are processed by freeze-drying technology.

21,000T

**Snack for Cat - 100gram**

100% organic fresh beef liver are processed by freeze-drying technology.

26,000T

## For dogs

**Snack for Dog - 60gram**

100% organic fresh beef liver are processed by freeze-drying technology.

21,000T

**Snack for Dog - 100gram**

100% organic fresh beef liver are processed by freeze-drying technology.

26,000T

© 2023 Nomadic Pet Care LLC. All rights reserved.

[info@nomadic-petcare.com](#)

Powered by [Snack-Partners](#)

For cats
For dogs
About Us
English
My Cart

## About us

Copy link

Watch on

## About Us

Nomadic Pet Care LLC was established in 2021 and operates in pet food production, foreign trading, and export sectors. We use freeze-drying technology to produce pet food that meets international standards.

## Product

We manufacture our products with fresh meat and milk of pasture-raised livestock which grazes on beautiful nature of Mongolia. In Mongolia, the horses live outdoors all year, dealing with temperatures from 30 °C (86 °F) in summer down to -40 °C (-40 °F) in winter, and horses can travel 150-500km per day. This Mongolian horse shows how organic the meat. Our factory has the capacity of 100 tons final products per year and exports products that meet international food standards.

## Partners

© 2023 Nomadic Pet Care LLC. All rights reserved.

[info@nomadic-petcare.com](#)

Powered by [Snack-Partners](#)

For cats
For dogs
About Us
English
My Cart

## For cats

**Snack for Cat - 60gram**

100% organic fresh beef liver are processed by freeze-drying technology.

21,000T

**Snack for Cat - 100gram**

100% organic fresh beef liver are processed by freeze-drying technology.

26,000T

© 2023 Nomadic Pet Care LLC. All rights reserved.

[info@nomadic-petcare.com](#)

Powered by [Snack-Partners](#)

For cats
For dogs
About Us
English
My Cart

## For cats

**Snack for Cat - 60gram**

100% organic fresh beef liver are processed by freeze-drying technology.

21,000T

**Snack for Cat - 100gram**

100% organic fresh beef liver are processed by freeze-drying technology.

26,000T

## Snack for Cat - 60gram

21,000T

1

[Add to cart](#)

**Digestion**

Vibrant skin and coat

**Stomach, Vitality and Weight management**

Teeth and Gums

**Palatability**

Moisture	Calories	Protein	Crude Fat	Crude Fiber	Ash
Max	Min	Min	Max	Max	Max
6.5%	408	71.5	4.0%	4.57%	1.5%

**Ingredients:**

Our products are free of chemical additives, 100% organic fresh beef liver are processed by freeze-drying technology.

**Benefits:**

- Beef liver is rich in vitamin A, B, magnesium, potassium, minerals, low fat, and high protein. It can prevent vitamin and protein deficiency in pets, reduce hair loss, reduce teeth and mouth diseases, and prevent muscle weakness.
- Ideal for pets who are overweight, diabetic, who have allergies or gastrointestinal disorders or pregnant and new mother pets.

**Feed tips:**

- Use pets that are more than three months old.

© 2023 Nomadic Pet Care LLC. All rights reserved.

[info@nomadic-petcare.com](#)

Powered by [Snack-Partners](#)

# Дүгнэлт

Миний бие Г. Зандан-Очир нь үйлдвэрлэлийн дадлагыг систем хөгжүүлэлт, клауд консалтинг зэргээр үйл ажиллагаа явуулдаг "Tech Partners ХХК"-нд ахлах хөгжүүлэгч Д. Бат-Оргилын удирдлага дор "Nomadic Pet Care LLC" компанийн бараа бүтээгдэхүүн борлуулах Ecommerce системийн фронт-энд хөгжүүлэлтийг хийж гүйцэтгэлээ. Дадлагын хугацаанд сервер дүрслэлт, динамик хуудас дүрслэл, цогц хүүк бичих, олон хэлээр API дуудах, JavaScript хэл дээр богино бөгөөд ойлгомжтой бичиглэл, CSS анимешн зэрэг Фронт-энд хөгжүүлэгчийн мэдэх шаардлагатай ур чадвар мэдлэгүүдийг олж авлаа. Мөн JavaScript хэлний эко систем дундаас хөгжүүлэгчийн туршлагыг сайжруулах, төвөгтэй үйлдлийг хөнгөвчлөх маш олон сан, фреймворкуудыг судалж хэрэглэж дадлаа.

Удирдагчаасаа хамгийн том сурсан зүйл нь асуудлыг инженер хүний нүдээр харж мэдэхгүй гэдэг дээр төвлөрөх биш асуудлаа ойлгож гарц гаргалгаа олдог, илүү сайн шийдэл судалж олж мэдэх, дэвшүүлдэг байх нь сайн инженерийг илтгэх шинж юм.

Цаашид фронт энд хөгжүүлэлтээ судлахаас гадна бусад back-end, Devops гээд хөгжүүлэлтийн бусад ойлголтуудыг судалж өгсөн үүрэг даалгавар, тулгарсан асуудал бүрийг шийдэж чадах инженер болох зорилготой байна.