

Quiz-pink

Лэндинги

| | |
|--------------------|---|
| Папка с лэндингами | https://gc.fitnessmama.co/pl/cms/page?folderId=248492 |
| Квиз | https://gc.fitnessmama.co/quiz https://gc.fitnessmama.co/pl/cms/page/editor?id=4564071 доп. поле offer_quiz_url |
| Офферы | https://gc.fitnessmama.co/offer https://gc.fitnessmama.co/pl/cms/page/editor?id=4564067 |

После прохождения квиза, ссылка с параметрами остается в этом доп поле

{deal_utm_group}

Далее его можно использовать в письмах или процессах по заказам

Виджеты

| Название | Виджет | Код вставки |
|----------------|---|---|
| авторизация | https://gc.fitnessmama.co/pl/lite/widget/editor?id=1602182 | <pre><script id="9d2530e680aa488cfd0688f886b298076ec76db5" src="https://gc.fitnessmama.co/pl/lite/widget/script?id=1602182"></script></pre> |
| 1 -3 -12 month | https://gc.fitnessmama.co/pl/lite/widget/editor?id=1602173 | <pre><script id="4a06808800bf64337e9548e7cc197bdfa1630a49" src="https://gc.fitnessmama.co/pl/lite/widget/script?id=1602173"></script></pre> |

Офферы

Технический заказ после прохождения квиза

Completed the pink quiz

<https://gc.fitnessmama.co/pl/sales/offer/update?id=8361463>

Архитектура и реализация

Лендинги `quiz-pink.html` и `offer.html` — самодостаточные одностраничные HTML-файлы (без сборки, фреймворков, npm). В GetCourse они залиты как страницы типа «Лендинг» (без штатной шапки/футера) через raw-HTML-блок. Контракт между ними — URL-параметры.

Файлы в проекте

| | |
|---|--|
| <code>quiz-pink.gc.html</code> | Версия квиза для GC. Залита на <code>/quiz</code> . |
| <code>offer.gc.html</code> | Версия оффер-страницы для GC. Залита на <code>/offer</code> . |
| <code>gc-widget-form.css</code> | Кастомный CSS для редактора формы 1602182 (email-шаг квиза). |
| <code>gc-widget-offer.css</code> | Кастомный CSS для редактора формы 1602173 (тарифы на <code>/offer</code>). |
| <code>script-widget-form.html</code> | JS-сниппет в HTML-блок формы 1602182 — валидирует поля + редиректит после успешного сабмита. |
| <code>quiz-pink.html</code> / <code>offer.html</code> | Оригиналы для standalone-тестирования (без GC-обёрток). |
| <code>DOCS.md</code> | Подробная техническая документация по UX-логике квиза и оффера (BMI-профили, weeks-формула, и т.п.). |

Изоляция от GC-системных стилей и JS

Контент каждого лендинга обернут в `<div class="fm-quiz">` / `<div class="fm-offer">`. Все CSS-селекторы префиксованы этим классом (164 правила в квизе, 293 в оффере) — это обеспечивает специфичность (0,2,0)+, которая бьёт GC-нормалайз (`h1 { font-size: 2em }`, (0,0,1)) и GC-typography (`#ltBlock... { font-family: Ubuntu !important }`).

JS обернут в IIFE, публичные хендлеры экспортируются на `window.FM_Quiz.*` и `window.FM_Offer.*`. Inline `onclick="fn()"` переписаны на `onclick="FM_Quiz.fn()"` (11 в квизе, 8 в оффере). Локальные переменные `name`, `height`, `body` в оффере переименованы в `userName`, `heightCm`, `bodyType` чтобы не пересекаться с глобалами браузера.

Шрифты (DM Sans, Syne) подключены через прямой `@font-face` на `fonts.gstatic.com` WOFF2 + `@import` как фоллбэк.

URL-контракт `quiz` → `offer`

После прохождения квиза браузер редиректит на `https://gc.fitnessmama.co/offer?...` с 13 параметрами:

`name`, `cw` (current weight), `tw` (target weight), `goal`, `age`, `diet`, `areas`, `freq`, `body`, `h` (height), `holding`, `energy`, `mot2`

`offer.gc.html` читает их через `new URLSearchParams(location.search)`. Все параметры имеют фоллбэк-дефолты, чтобы страница работала standalone.

Формула числа недель (продублирована в обоих файлах, должна совпадать):

```
Math.max(Math.ceil(Math.abs(cw - tw) / 0.65), 4)
```

Интеграция виджетов GetCourse

Email-шаг квиза и блок с тарифами на оффер-странице заменены на GC-виджеты (форма 1602182 и 1602173 соответственно). Виджеты грузятся как кросс-доменные iframe и сохраняют лиды/заказы напрямую в CRM GetCourse.

Монтаж виджета в квизе (форма 1602182)

В функции `buildEmail()` вместо HTML-полей `name/email` рендерится `<div id="gc-form-mount">`.
Функция `mountGcForm()`:

1. Через `history.replaceState` пушит ответы квиза в URL родительской страницы (`/quiz?name=...&cw=...&...`). Виджет-iframe автоматически наследует `window.location.search` в свой `src`.
2. Программно создаёт `<script id="9d2530..." src=".../widget/script?id=1602182">` (`innerHTML` не запускает скрипты — нужен `document.createElement`).
3. В `script.onload` диспатчит кастомный event `StartWidget9d2530...` — без него виджет не запустится, потому что `DOMContentLoaded` уже отработал.

Монтаж виджета в оффере (форма 1602173)

На `/offer` виджет смонтирован **в двух местах**: верхняя секция `#pricing-top` и нижняя `#pricing`. Скрипт виджета GC поддерживает только одну инстанцию (привязан к захардкоженному id и удаляет себя после первого запуска), поэтому в коде создаются **два iframe напрямую** в обход скрипт-обёртки. Один глобальный `postMessage-listener` слушает `e.data.uniName === '4a06808800bf64337e9548e7cc197bdfa1630a49'` и одновременно обновляет высоту обоих iframe.

Обходы GC-скоупинга CSS

GetCourse автоматически дописывает `#ltBlock<id>` после каждой запятой и закрывающей `}` в кастомном CSS формы. Это ломает несколько типичных паттернов:

1. **Одиночные правила на `body` и `html`** превращаются в `#ltBlock... body { ... }`, который не матчит ничего (ID находится *внутри* `body`). **Обход:** ставить `body` или `html` **первым в списке через запятую** с junk-селектором — GC оставляет первый селектор нетронутым: `body, ._noop1 { ... }` → `body, #ltBlock... ._noop1 { ... }`.
2. **Padding на `body` не учитывается в высоте `iframe`** — GC меряет высоту через `$(window.document.body).height()` (jQuery, исключает padding). **Обход:** ставить padding на `.lt-block-wrapper` (вложенный элемент), его `outerHeight` попадает в `body.height()`.
3. **Bootstrap-класс `.clearfix` на `.form-position`** добавляет невидимые `::before / ::after` с `display: table`. В flex-раскладке они становятся flex-итемами и сдвигают радио-кнопки. **Обход:** `label.form-position::before, label.form-position::after { display: none !important; content: none !important; }`, а для MOST POPULAR-баджа на 3-month переопределить `::before` с `position: absolute`.
4. **Высокоприоритетный GC-шрифт 'Ubuntu'** ставится на `#ltBlock` с `!important` и распространяется через наследование. **Обход:** поставить `font-family: 'DM Sans' !important` на `.lt-block-wrapper` — оборвёт цепочку наследования для всех потомков формы.
5. **Запятые в `@import URL` и в `src: url(...woff2), url(...ttf)`** ломаются GC-парсером. **Обход:** использовать одиночный `src` без fallback (WOFF2 поддерживается всеми современными браузерами).

JS-логика редиректа после формы (script-widget-form.html)

Нативного макроса для проброса URL-параметров после сабмита у GC нет (`{query_string}` не подставляется). Поэтому в HTML-элемент формы 1602182 вставлен кастомный скрипт со следующей логикой:

1. **Клиентская валидация перед сабмитом.** При клике submit-кнопки функция `isFormValid()` проверяет:
 - `input[name="formParams[first_name]"]` — имя не пустое
 - `input[name="formParams[email]"]` — email соответствует regex
 - `input.global-confirm-checkbox` — Privacy Policy чекбокс отмеченЕсли что-то не пройдено — флаг `pending` остаётся `false`, редирект не сработает даже если AJAX отвлится.
2. **Перехват только нужного XHR.** Через `$(document).ajaxComplete` ловим запросы и фильтруем по `settings.url.indexOf('block-public/process')` — это эндпоинт самой формы. Любые посторонние XHR (метрики, аналитика) игнорируются — иначе они триггерили бы редирект *до* того, как сама форма успеет создать заказ.
3. **Подтверждение успеха.** Парсим `xhr.responseText`, проверяем `r.success === true`. Только после этого редиректим — на этом моменте GC уже создал лид/заказ в CRM.
4. **Чистка query string.** Из `window.location.search` `iframe` удаляются системные GC-параметры (`id`, `ref`, `loc`, `gcSession`, `gcVisitor`, `gcVisit`, `gcSessionHash`, `clrtQueryData`, `sid`, `sfix`), остаются только ответы квиза. Итоговый URL:

`https://gc.fitnessmama.co/offer?name=...&cw=...&tw=...&...`

5. **Выход из iframe.** Редирект через `window.top.location.href` — переводит родительское окно, а не сам iframe.

Отладка

Скрипт логирует все этапы в консоль iframe виджета. Чтобы увидеть лог — DevTools → переключиться на iframe (Chrome: dropdown «top» → выбрать

`gc.fitnessmama.co/pl/lite/widget/widget?...`) → Console. Ожидаемая цепочка после клика:

```
[FM redirect] submit clicked, форма валидна, ждём ответ
```

```
[FM redirect] form submit response status 200
```

```
[FM redirect] → https://gc.fitnessmama.co/offer?name=...&cw=...&tw=...
```

Если валидация не прошла: `[FM redirect] валидация не пройдена — редирект не будет: {nameFilled, emailValid, privacyChecked}`.

Визуальные доработки тарифов через CSS-псевдоэлементы

Поскольку HTML-разметку формы GC мы менять не можем, бейджи **MOST POPULAR / SAVE X% / €X/day** добавлены через `::before` и `::after` с `content` по `data-offer-id`:

| data-offer-id | Тариф | SAVE | Per-day | MOST POPULAR |
|----------------------|----------|------|-----------|--------------|
| <code>7334808</code> | 1-Month | 60% | €0.83/day | — |
| <code>7335866</code> | 3-Month | 70% | €0.70/day | ✓ |
| <code>7335868</code> | 12-Month | 80% | €0.41/day | — |

Значения захардкожены в `gc-widget-offer.css` (псевдоэлементы не умеют считать математику). Если цены или offer-id поменяются — нужно обновить вручную.

Чек-лист при изменении

- Поменялся offer-id плана → обновить селекторы `label.form-position[data-offer-id="..."]` в `gc-widget-offer.css`.
- Поменялась цена или длительность плана → пересчитать SAVE % и €/day в CSS (`(old-new)/old`, `new/duration`).
- Добавили новое поле в форму GC → добавить класс в селектор-список `input.f-input, textarea.f-input, ...` в соответствующем CSS.
- Поменялся URL offer-страницы → правка константы `OFFER_URL` в `quiz-pink.gc.html` и в `buildRedirectUrl()` внутри `script-widget-form.html`.
- После любой правки CSS — в редакторе формы GC: *Сохранить* → *Опубликовать*, потом hard-reload страницы (Cmd+Shift+R).

Revision #9

Created 11 May 2026 12:08:02 by David

Updated 13 May 2026 22:42:28 by David