

# 투어비스 공통 웹 컴포넌트 사용법

## 공통 가이드

### 1) 위젯 스토리북

[위젯 스토리북 보기](#)

### 2) 리액트 타입스크립트 가이드(widget.d.ts)

```
1 // Typescript(웹컴포넌트 타입 선언)
2 import { HTMLAttributes } from 'react';
3
4 import { OmakaseInventory } from './omakase-inventory';
5
6 declare global {
7   namespace JSX {
8     interface TvPayWidgetProps extends HTMLAttributes {
9       'order-code': string;
10       env: string;
11       payment: boolean;
12       refund: boolean;
13     }
14
15     interface HamburgerMenuWidgetProps extends HTMLAttributes {
16       env?: string;
17       open: boolean;
18       login: boolean;
19       name: string;
20       coupon: number;
21       club?: boolean;
22       company?: string;
23     }
24
25     interface OmakaseuWidgetProps extends HTMLAttributes {
26       env?: string;
27       inventory: OmakaseInventory;
28       city?: string;
29       hotel?: string;
30       product?: string;
31       air?: string;
32       'divide-top'?: boolean;
33       'divide-bottom'?: boolean;
34     }
35
36     interface HotelWidgetProps extends HTMLAttributes {
37       env: string;
38     }
39 }
```

```
40 interface TvTntProductWidgetProps extends HTMLAttributes {
41   id: string;
42   env: 'LOCAL' | 'DEV' | 'STG' | 'PRD';
43   type: 'cart' | 'order';
44 }
45
46 interface TvTntRefundPolicyWidgetProps extends HTMLAttributes {
47   id: string;
48   env: 'LOCAL' | 'DEV' | 'STG' | 'PRD';
49   type: 'cart' | 'order';
50 }
51
52 interface GNBWidgetProps extends HTMLAttributes {
53   env?: string;
54   category: string;
55   pc?: boolean;
56   sticky?: boolean;
57 }
58
59 interface FooterWidgetProps extends HTMLAttributes {
60   env?: string;
61   extend?: boolean;
62   pc?: boolean
63 }
64
65 interface CategoryQuickSearchWidgetProps extends HTMLAttributes {
66   env?: string;
67   category: string;
68   gnbCategory: string;
69 }
70
71 interface LoadingWidgetProps extends HTMLAttributes {
72   message: string;
73 }
74
75 interface BottomTabBarWidgetProps extends HTMLAttributes {
76   env?: string;
77 }
78
79 interface IntrinsicElements {
80   'tv-pay-widget': TvPayWidgetProps;
81   'hamburger-menu-widget': HamburgerMenuWidgetProps;
82   'omakase-widget': OmakaseuWidgetProps;
83   'tv-tnt-product-info-widget': TvTntProductWidgetProps;
84   'tv-tnt-refund-policy-widget': TvTntRefundPolicyWidgetProps;
85   'hotel-search-widget': HotelWidgetProps;
86   'gnb-widget': GNBWidgetProps;
87   'footer-widget': FooterWidgetProps;
88   'category-quick-search-widget': CategoryQuickSearchWidgetProps;
89   'loading-widget': LoadingWidgetProps;
90   'bottom-tab-bar-widget': BottomTabBarWidgetProps;
91 }
92 }
93 }
```

## 공통 위젯

### 1) GNB

- script url

- dev + stg : <https://d1ad4gpy4661in.cloudfront.net/tourvis-static/common/common-widget.js>
- prod : <https://d2um1hurm6o2hd.cloudfront.net/tourvis-static/common/common-widget.js>

- 예제

```
1 <script src="위에 스크립트 URL"/>
2 <gnb-widget
3   env={'development'}
4   category={'tnt'}
5   device={'pc'}
6   sticky={false}
7   blank={false}
8 />
```

- API

- a. props (attr)

- i. env 'development' | 'stage' | 'production' - 구동 서버의 환경변수 (필수)

- ii. category 'common' | 'air' | 'hotel' | 'tnt' | 'package' | 'blank' 각 도메인 별 카테고리 (필수)  
blank - 로고만 있는 GNB의 경우

- iii. pc boolean - PC 전용 페이지인 경우 필수, 기본값 false

- iv. sticky boolean - sticky를 해제하고 싶은 경우에만 false, 기본값 true (선택)

- v. blank boolean - 로고만 있는 GNB의 경우 true, 기본값 false (선택)

- b. 커스텀 이벤트

- i. modal-open: GNB에서 최근본여행 팝업 모달이 오픈된 때 이벤트가 발생한다.
- ii. modal-close: GNB에서 최근본여행 팝업 모달이 닫힐 때 이벤트가 발생한다.
- iii. 사용 케이스 예시: 팝업 모달이 오픈되어도 뒤 배경 부분에 스크롤이 가능하게 되는 문제 등의 팝업 모달이 떠 있을 때 GNB를 사용하는 페이지에서 배경 화면의 스크롤 방지.
- iv. 예제 코드(React)

```
1  useEffect(() => {
2    const handleModalOpen = () => {
3      const body = document.getElementsByTagName("body");
4      if (!!body[0] && !body[0].classList.contains('modal-open')) {
5        body[0].classList.add('modal-open');
6      }
7    };
8
9    const handleModalClose = () => {
10     const body = document.getElementsByTagName("body");
11     if (!!body[0] && body[0].classList.contains('modal-open')) {
12       body[0].classList.remove('modal-open');
13     }
14   };
15
16   window.addEventListener('modal-open', handleModalOpen);
17   window.addEventListener('modal-close', handleModalClose);
18
19   return () => {
20     window.removeEventListener('modal-open', handleModalOpen);
21     window.removeEventListener('modal-close', handleModalClose);
22   };
23 }, []);
```

- 로딩 상태에 따라 빈 div로 layout shift 막아주시면 됩니다.

▼ 리액트 가이드

```
1  // GNB
2  'use client';
3
4  import { useWebComponent } from '../hook';
5
6  interface IGNBProps {
7    category?: 'common' | 'air' | 'hotel' | 'tnt' | 'package' | 'blank';
8    isForPC?: boolean;
9  }
10
```

```

11  /**
12   * @prop {string} category 카테고리명
13   * @prop {boolean} isForPC PC 고정 여부
14   */
15  export function GNB({ category = 'common', isForPC = false }: IGNBProps) {
16    const { isLoading } = useWebComponent('gnb-widget');
17
18    if (!isLoading) {
19      return (
20        <nav style={{ minHeight: category === 'blank' ? '87px' : '137.4px' }} />
21      );
22    }
23
24    return (
25      <gnb-widget
26        env={process.env.NEXT_PUBLIC_APP_ENV}
27        category={category}
28        device={isForPC ? 'pc' : ''}
29      />
30    );
31  }
32
33  export default GNB;
34
35  // useWebComponent
36  import { universalEnv } from '../env/universal-env';
37  import { WebComponents } from '../types/web-component/web-component';
38  import useScriptLoader from './useScriptLoader';
39
40  const WebComponentSrcMap: Record<WebComponents, string> = {
41    'gnb-widget': `${universalEnv.scriptUrls.assetPrefix}/common/common-widget.js`,
42  };
43
44  export const useWebComponent = (component: WebComponents) => {
45    const { isLoading, isError, isLoading } = useScriptLoader(
46      WebComponentSrcMap[component],
47      'body',
48      () => !!customElements.get(component),
49    );
50
51    return { isLoading, isError, isLoading };
52  };
53
54  export default useWebComponent;
55
56  import { useEffect, useState } from 'react';
57
58  const useScriptLoader = (
59    scriptSrc: string,
60    scriptTarget: 'head' | 'body',
61    earlyReturnCheck: () => boolean,
62    onLoadCb?: () => void,
63    onErrorCb?: () => void,
64  ) => {
65    const [isLoading, setIsLoading] = useState(true);
66    const [isError, setIsError] = useState(false);
67
68    const isLoading = !isLoading && !isError;

```

```

69
70 useEffect(() => {
71   if (earlyReturnCheck()) {
72     setIsLoading(false);
73     return;
74   }
75
76   // 스크립트 등록
77   const script = document.createElement('script');
78   script.src = scriptSrc;
79   script.async = true;
80   script.onload = () => {
81     setIsLoading(false);
82     onLoadCb && onLoadCb();
83   };
84   script.onerror = err => {
85     console.error(err);
86     setIsError(true);
87     onErrorCb && onErrorCb();
88   };
89
90   document[scriptTarget].appendChild(script);
91
92   return () => {
93     if (!earlyReturnCheck()) {
94       document[scriptTarget].removeChild(script);
95     }
96   };
97 }, []);
98
99 return { isLoading, isError, isLoaded };
100 };
101
102 export default useScriptLoader;
103

```

## 1-1] GNB - 퀵서치 (별도 이용시)

### • 예제

```

1 <!-- 일반 등록 가정 -->
2 <script src="위에 스크립트 URL"/>
3 <div
4   id="quick-search"
5   data-env={"development"}
6 />
7 <!-- ex: register add -->
8 <script type="application/javascript">
9   window.onload = function () {
10     window.registerQuickSearch(); // register 함수
11   };
12 </script>
13
14
15

```

### • 리액트의 경우 위의 과정 참고

- ```

1  export function QuickSearch() {
2    const { isLoading } = useWebComponent('quick-search-widget');
3
4    if (!isLoading) {
5      return (
6        <div></div>
7      );
8    }
9
10   return (
11     <quick-search-widget
12       env={process.env.NEXT_PUBLIC_APP_ENV}
13       category={category}
14     />
15   );
16 }
```

- API

- a. props (attr)

- i. data-env 'development' | 'stage' | 'production' - 구동 서버의 환경변수 (필수)

- ii. data-category 'air' | 'hotel' | 'tnt' | 'package' | 'place' 각 탭 별 활성화 (선택)

## 2] Footer

- script url

- dev + stg : <https://d1ad4gpy4661in.cloudfront.net/tourvis-static/common/common-widget.js>
  - prod : <https://d2um1hurm6o2hd.cloudfront.net/tourvis-static/common/common-widget.js>

- 예제

```

1  <script src="위에 스크립트 URL"/>
2  <footer-widget
3    env={'development'}
4    extend={false}
5    pc={true}
6  />
```

- API

- a. props (attr)

- i. env `'development' | 'stage' | 'production'` - 구동 서버의 환경변수 (필수)
- ii. extend `boolean` 각 카테고리별 메인 페이지에서 확장 푸터 사용 여부 (선택) - 메인 페이지에서는 필수
- iii. pc `boolean` - PC 전용 페이지인 경우 필수, 기본값 `false`

### 3) 카테고리 퀵서치

- script url
  - dev + stg : <https://d1ad4gpy4661in.cloudfront.net/tourvis-static/common/common-widget.js>
  - prod : <https://d2um1hur6o2hd.cloudfront.net/tourvis-static/common/common-widget.js>
- 예제

```
1 interface CategoryQuickSearchProps {
2   category: 'main' | 'search' | 'product';
3   gnbCategory: 'common' | 'air' | 'hotel' | 'tnt' | 'package' | 'blank';
4 }
5
6 /**
7  * @prop {string} category 카테고리명
8  * @prop {string} gnbCategory gnb카테고리명
9  */
10 export default function CategoryQuickSearch({
11   category,
12   gnbCategory,
13 }: CategoryQuickSearchProps) {
14   const { isLoading } = useWebComponent('category-quick-search-widget');
15
16   if (!isLoading) {
17     return (
18       <div className="quick-search-panel" style={{ minHeight: 157.5 }}></div>
19     );
20   }
21
22   // GNB 및 tourvis-front와 z-index 호환위해 꼭 필요
23   const css = `
24     category-quick-search-widget {
25       margin: 0;
26       z-index: 990;
27     }`;
28 }
```



```

29 return (
30   <>
31   <style>{css}</style>
32   <category-quick-search-widget
33     env={process.env.NEXT_PUBLIC_APP_ENV}
34     category={category}
35     gnbCategory={gnbCategory}
36   />
37 </>
38 );
39 }

```

## • API

### a. props (attr)

- env 'development' | 'stage' | 'production' - 구동 서버의 환경변수 (필수)
- category 'main' | 'search' | 'product'; 각 탭 별 활성화 (필수)
- gnbCategory 'common' | 'air' | 'hotel' | 'tnt' | 'package' | 'black; 각 탭 별 활성화 (필수)

## • 기타

- 카테고리 퀵서치 컴포넌트의 경우 GNB와의 sticky 관계때문에 GNB의 sticky props와 같이 사용해야함.
- 현재 투티의 경우 GNB의 sticky 값을 false로 하고 category props가 main이 아닌경우에만 sticky가 활성화 됨
- gnbCategory는 해당 GNB의 height 계산을 위해 필요.

## 4) 로딩 위젯(PC/MO 공통)

### • script url

- dev + stg : <https://d1ad4gpy4661in.cloudfront.net/tourvis-static/common/common-widget.js>
- prod : <https://d2um1hur6o2hd.cloudfront.net/tourvis-static/common/common-widget.js>

## • 예제

```

1 // 스크립트 연결
2 <script src="위에 스크립트 URL"/>
3

```

4 <loading-widget message="예약상세 화면을 불러오는 중입니다." />

- API

- a. props (attr)

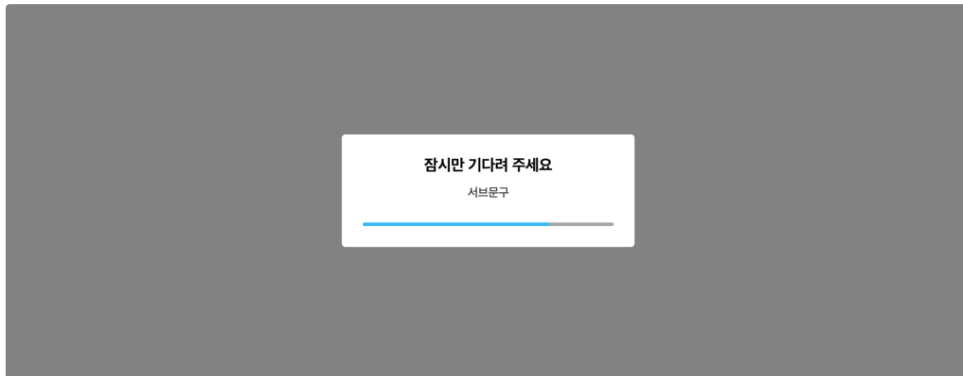
- message: 로딩 팝업에 표시할 텍스트 (필수)

- 기타

- 로딩 팝업 뒷배경의 스크롤 차단과 같은 부가 기능은 사용자가 직접 구현해야 함.
  - 예제 (React)

```
1 useEffect(() => {
2   const body = document.getElementsByTagName('body');
3
4   if (!!body[0] && !body[0].classList.contains('modal-open')) {
5     body[0].classList.add('modal-open');
6   }
7
8   return () => {
9     if (!!body[0] && body[0].classList.contains('modal-open')) {
10      body[0].classList.remove('modal-open');
11    }
12  };
13 }, []);
```

- 실행화면



- 

## 5) 하단 탭바 (MO)

- script url

- dev + stg : <https://d1ad4gpy4661in.cloudfront.net/tourvis-static/common/common-widget.js>
  - prod : <https://d2um1hur6o2hd.cloudfront.net/tourvis-static/common/common-widget.js>

- 예제

```
1 <script src="위에 스크립트 URL"/>
2 <bottom-tab-bar-widget
3   env={'development'}
4 />
```

- API

- a. props (attr)

- i. env 'development' | 'stage' | 'production' -  
구동 서버의 환경변수 (필수)

## 6) 앱 다운로드 유도 (MO)

- script url

- dev + stg : <https://d1ad4gpy4661in.cloudfront.net/tourvis-static/common/common-widget.js>
  - prod : <https://d2um1hur6o2hd.cloudfront.net/tourvis-static/common/common-widget.js>

- 예제

```
1 <script src="위에 스크립트 URL"/>
2 <app-down-banner-widget
3   env={'development'}
4   target='comm_main'
5 />
```

- API

- a. props (attr)

- i. env 'development' | 'stage' | 'production' -  
구동 서버의 환경변수 (필수)
    - ii. target - 사용하는 위치 (필수)

```
1 export type AppDownloadBannerTarget =
2   | 'comm_main'
3   | 'mypage'
4   | 'reservation'
5   | 'htl_main'
6   | 'htl_detail'
7   | 'tnt_main'
8   | 'tnt_detail'
9   | 'pkg_main'
10  | 'pkg_detail'
```

## 99] 햄버거 메뉴

- script url
  - dev + stg : <https://d1ad4gpy4661in.cloudfront.net/tourvis-static/common/common-widget.js>
  - prod : <https://d2um1hur6o2hd.cloudfront.net/tourvis-static/common/common-widget.js>

## • 예제

```

1 // 스크립트 연결
2 <script src="위에 스크립트 URL"/>
3
4 <hamburger-menu-widget
5   env={process.env.NEXT_PUBLIC_APP_ENV}
6   open={menuVisible}
7   login={!user}
8   name={user?.cMbrNm}
9   coupon={user?.couponIssueCount}
10  club={user?.isClubMember}
11  company={user?.companyName}
12 />

```

## • API

- 웹 컴포넌트 이름은 `hamburger-menu-widget` 입니다.
- props (attr)
  - env `'development' | 'stage' | 'production'` - 구동 서버의 환경변수
  - open `boolean` - 햄버거 메뉴 열림 여부
  - login `boolean` - 로그인 여부
  - name `string` - 로그인한 유저의 이름
  - coupon `number` - 로그인한 유저의 쿠폰 갯수

- vi. club **boolean** : 로그인한 유저의 클럽 투어비스 여부
- vii. company **string** : 유저 api 내부 entpNm값 (기업회원).

### c. 이벤트 리스너

웹 컴포넌트 내부에서는 외부 사용처 (React, Next 등)의 상태를 직접적으로 제어할 수 없습니다. 그에 따라 커스텀 이벤트를 통해 상태를 제어해줘야합니다. 웹 컴포넌트 내부에서는 전역 window 객체에 **hamburger-menu-close** 라는 이름으로 이벤트를 발생시킵니다. 해당 이벤트를 감지해서 **open** 상태를 바꿔주는 함수를 연결해주시면 됩니다.

```
1  useEffect() => {  
2    const handleMenuClose = () => {  
3      setMenuVisible(false);  
4    };  
5  
6    window.addEventListener('hamburger-menu-close', handleMenuClose);  
7  
8    return () => {  
9      window.removeEventListener('hamburger-menu-close', handleMenuClose);  
10   };  
11 }, []);
```

---

## 오마카세 위젯

### 1) 오마카세 모바일

- script url
  - dev + stg : <https://d1ad4gpy4661in.cloudfront.net/tourvis-static/common/omakase-widget.js>
  - prod : <https://d2um1hur6o2hd.cloudfront.net/tourvis-static/common/omakase-widget.js>

- 예제

```
1 // 스크립트 연결
```

```

2 <script src="위에 스크립트 URL"/>
3
4 <omakase-widget
5   env={process.env.NEXT_PUBLIC_APP_ENV}
6   inventory={"mo_common_main_b"}
7 />

```

## • API

a. 웹 컴포넌트 이름은 **omakase-widget** 입니다.

b. props (attr)

i. env **'development' | 'stage' | 'production'** - 구동 서버의 환경변수 (필수)

ii. inventory **'mo\_common\_main\_a' | ....** - 오마카세 mo inventory (필수)

iii. city - **city\_master\_id** API 요청시 해당 인벤토리가 필요한 값만 추가해주시면 됩니다.

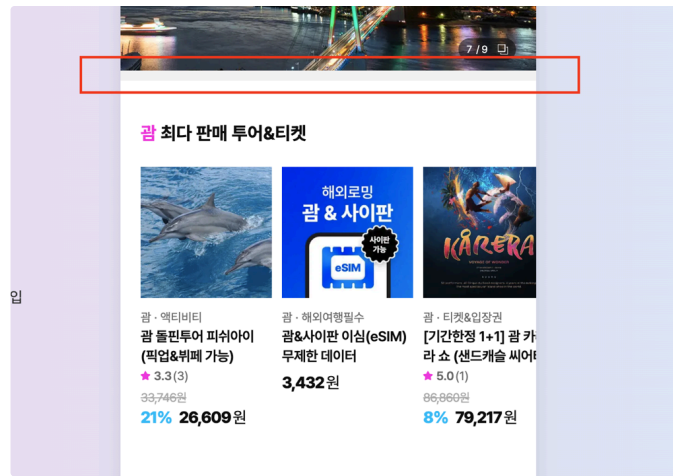
iv. hotel - **htl\_master\_id** API 요청시 해당 인벤토리가 필요한 값만 추가해주시면 됩니다.

v. product - **product\_id** API 요청시 해당 인벤토리가 필요한 값만 추가해주시면 됩니다.

vi. air - **iata\_airport\_code** API 요청시 해당 인벤토리가 필요한 값만 추가해주시면 됩니다.

vii. divide-top **boolean** - 오마카세 상단 divide 영역 필요 여부

viii. divide-bottom **boolean** - 오마카세 하단 divide 영역 필요 여부



## 2) 오마카세 PC

- script url

- dev + stg : <https://d1ad4gpy4661in.cloudfront.net/tourvis-static/common/pc-omakase-widget.js>
- prod : <https://d2um1hurm6o2hd.cloudfront.net/tourvis-static/common/pc-omakase-widget.js>

- 예제

```

1 // 스크립트 연결
2 <script src="위에 스크립트 URL"/>
3
4 <pc-omakase-widget
5   env={process.env.NEXT_PUBLIC_APP_ENV}
6   inventory={"pc_common_main_b"}
7 />

```

- API

a. 웹 컴포넌트 이름은 **omakase-widget** 입니다.

b. props (attr)

i. env **'development' | 'stage' | 'production'** - 구동 서버의 환경변수 (필수)

ii. inventory **'pc\_common\_main\_a' | ....** - 오마카세 pc inventory (필수)

- iii. city - `city_master_id` API 요청시 해당 인벤토리가 필요한 값만 추가해주시면 됩니다.
- iv. hotel - `htl_master_id` API 요청시 해당 인벤토리가 필요한 값만 추가해주시면 됩니다.
- v. product - `product_id` API 요청시 해당 인벤토리가 필요한 값만 추가해주시면 됩니다.
- vi. air - `iata_airport_code` API 요청시 해당 인벤토리가 필요한 값만 추가해주시면 됩니다.

## 기타 위젯

### 1) 투어 & 티켓 구매할 상품 정보

- script url

- dev + stg : <https://d1ad4gpy4661in.cloudfront.net/tourvis-static/common/tnt-product-info-widget.js>
- prod : <https://d2um1hurm6o2hd.cloudfront.net/tourvis-static/common/tnt-product-info-widget.js>

- 예제

```
1 // 스크립트 연결
2 <script src="위에 스크립트 URL"/>
3
4 <tv-tnt-product-info-widget
5   id="12345"
6   env="DEV"
7   type="cart"
8 />
```

- API

- a. 웹 컴포넌트 이름은 `tv-tnt-product-info-widget` 입니다.
- b. 투어비스 통합 결제 페이지에서 투티의 상품 정보를 제공할 목적으로 개발되었습니다.
- c. props (attr)



- i. id( `string` ): 첫결제 시에는 cartId를 입력합니다. 재결제 또는 추가결제 시에는 예약Id를 입력합니다.(필수)
- ii. env( `'LOCAL' | 'DEV' | 'STG' | 'PRD'` ): 서버 환경을 설정합니다. 로컬(LOCAL), 개발(DEV), 스테이지(STG), 운영(PRD). 환경 세팅에 따라서 연결되는 API 주소가 변경됩니다. (필수)
  - 1. LOCAL은 위젯 개발자가 로컬에서 확인용으로만 사용합니다.
- iii. type( `'cart' | 'order'` ): 첫결제 시에는 'cart', 재결제 또는 추가결제 시에는 'order'를 입력합니다.(필수)
- d. 연결 API 정보
  - i. 투어비스 투티 프론트 서버에서 제공하는 API를 사용합니다.
  - ii. 첫결제 화면: `https://{각 환경 호스트}/activity/api/v2/payment-cart/{cartId}`
  - iii. 재결제 화면 : `https://{각 환경 호스트}/activity/api/v2/payment-order/{orderId}`

## 2) 투어 & 티켓 구매할 상품 환불 정책

- script url
  - dev + stg : <https://d1ad4gpy4661in.cloudfront.net/tourvis-static/common/tnt-refund-policy-widget.js>
  - prod : <https://d2um1hurm6o2hd.cloudfront.net/tourvis-static/common/tnt-refund-policy-widget.js>
- 예제

```
1 // 스크립트 연결
2 <script src="위에 스크립트 URL"/>
3
4 <tv-tnt-refund-policy-widget
5   id="12345"
6   env="DEV"
```

```
7 type="cart"
8 />
```

- API

- a. 웹 컴포넌트 이름은 `tv-tnt-refund-policy-widget` 입니다.

- b. 투어비스 통합 결제 페이지에서 투티의 상품 환불 정책을 제공할 목적으로 개발되었습니다.

- c. props (attr)

- i. id( `string` ): 첫결제 시에는 `cartId`를 입력합니다. 재결제 또는 추가결제 시에는 `예약Id`를 입력합니다.(필수)

- ii. env( `'LOCAL' | 'DEV' | 'STG' | 'PRD'` ): 서버 환경을 설정합니다. 로컬(LOCAL), 개발(DEV), 스테이지(STG), 운영(PRD). 환경 세팅에 따라서 연결되는 API 주소가 변경됩니다. (필수)
      - 1. LOCAL은 위젯 개발자가 로컬에서 확인용으로만 사용합니다.

- iii. type( `'cart' | 'order'` ): 첫결제 시에는 'cart', 재결제 또는 추가결제 시에는 'order'를 입력합니다.(필수)

- d. 연결 API 정보

- i. 투어비스 투티 프론트 서버에서 제공하는 API를 사용합니다.

- ii. 첫결제 화면: `https://{각 환경 호스트}/activity/api/v2/payment-cart{cartId}`

- iii. 재결제 화면: `https://{각 환경 호스트}/activity/api/v2/payment-order/{orderId}`

### 3) 호텔 검색

- script url

- dev + stg : <https://d1ad4gpy4661in.cloudfront.net/tourvis-static/common/hotel-search-widget.js>
- prod : <https://d2um1hur6o2hd.cloudfront.net/tourvis-static/common/hotel-search-widget.js>

## • 예제

```
1 // 스크립트 연결
2 <script src="위에 스크립트 URL"/>
3
4 <hotel-search-widget
5   env="development"
6 />
```

## • API

- 웹 컴포넌트 이름은 `hotel-search-widget` 입니다.
- 투어비스 통합 검색 페이지를 위해 만듦.
- props (attr)
  - env `'development' | 'stage' | 'production'` - 구동 서버의 환경변수 (필수)
  - target `'gnb' | 'pc'` - 사용할 대상 (모바일은 빈값)
  - pathname `'hotel'` - 호텔 내부 여부
  - pagetype `'city' | 'detail'` - 호텔 내부 페이지 구분
  - pageinfo `'{"keyword": "서울", "in": "2024-02-28", "out": "2024-02-29" .....}'` - 호텔 내부 페이지 사용시 필요 파라미터 (json 형태의 string)
  - open `'true' | 'false'` 팝업 오픈 여부
  - opentype `'A' | 'B' | 'C'` 팝업 오픈 타입
- 연결 API 정보
  - 투어비스 호텔 파라곤 서버에서 제공하는 API를 사용합니다. `{ dhotelapi / stg-hotelapi / hotelapi }`

(CORS 에러로 Proxy 미들 서버 사용)

1. 휴일 조회: `/m/api/proxy-`

`hotel/v2/common/service-`  
`info/holidays`

2. 검색 조회: `/m/api/proxy-hotel-`  
`wc/search/tourviscomplete`

ii. 최근 검색 조회/저장 API

1. 최근 검색 API 적용

#### 4) 패키지 검색

- script url

- dev + stg : <https://d1ad4gpy4661in.cloudfront.net/tourvis-static/common/package-components.js>
- prod : <https://d2um1hurm6o2hd.cloudfront.net/tourvis-static/common/package-components.js>

- 예제

```
1 // 스크립트 연결
2 <script src="위에 스크립트 URL"/>
3
4 // Quick Search
5 <package-quick-search
6   env="development"
7 />
8
9 // GNB / Mobile Tab Search
10 <package-integrated-search
11   env="development"
12 />
```

- API

- a. 웹 컴포넌트 이름은 `package-quick-search`, `package-integrated-search` 입니다.
- b. 투어비스 통합 검색 페이지를 위해 만듦.

c. props (attr)

i. env 'development' | 'stage' | 'production' -

구동 서버의 환경변수 (필수)

d. 연결 API 정보

i. 휴일 API

1. 휴일 조회: /m/api/proxy-hotel/v2/common/service-info/holidays

ii. 패키지 검색 API <https://dev-package-api.tidesquare.com/swagger-ui.html?urls.primaryName=02%20Tourvis%20API#/04.%ED%80%B5%EC%84%9C%EC%B9%98%EC%A1%B0%ED%9A%8C>

1. 키워드 검색 조회:

/api/v1/tourvis/quick/keyword

2. 주요지역 조회:

/api/v1/tourvis/quick/place

3. 출발지 조회:

/api/v1/tourvis/quick/departure

iii. 최근 검색 조회/저장 API

1. 최근 검색 API 적용

배포 이력

• 2024-06-21

- [🔗DT-29358: \[투어비스\] 2024 2차 숙세페 GNB 배지 3만원 > 2만원 변경](#) 완료
- 2024-06-10
  - [🔗DT-28647: \[투어비스\] 내 여행 오마카세 : 플레이스 토픽 디자인 가이드 개선\\_F/E 개발](#) 완료
  - [☑DT-28702: \[투어비스\] 개발서버 모바일웹\\_원링크 수정](#) 완료
- 2024-05-30
  - [🔗DT-23789: \[투어비스\] 나의 여행 - MO > 하단 탭 바 > 메뉴 > 랜딩 페이지 변경](#) 완료
  - [🔗DT-26922: \[투어비스\] 나의 여행 - "PC - GNB" 및 "MO - 하단 탭" 내 버튼명 수정 요청](#) 완료
  - [🔗DT-27370: \[투어비스\] 오마카세 : 내 여행 오마카세 > 플레이스 토픽 위젯 개발](#) 완료
- 2024-05-28
  - [🔗DT-27810: \[투어비스\] 2024 2차 숙세페 개발\\_GNB\(공통\\_2구간\)](#) 완료
- 2024-05-21
  - [🔗DT-27723: \[투어비스\] 2024 2차 숙세페 개발\\_GNB\(공통\\_1구간\)](#) 완료
- 2024-04-29
  - [🔗DT-27011: \[투어비스\] 상단 배너 링크 수정](#) 완료
- 2024-04-25
  - [☑DT-26708: \[투어비스/개발\] PC > Footer 폰트/새창 추가 요청 건](#) 완료
  - [🔗DT-26768: \[투어비스\] 플레이스 '여행지 찾기' > 여행지 생성 요청 > GA 로그 반영 요청](#) 완료
  - [🔗DT-26804: \[투어비스\] 메인, 투액 배너에 ga 작업 요청](#) 완료
  - [🔗DT-25170: \[투어비스\] 모바일웹 접속자에게 노출되는 상단 배너 클릭시 원링크 적용\\_메인, 투액](#) 완료

- 2024-04-23

- ☒ DT-26707: [투어비스] MO > 하단탭바 > 카테고리 내 검색 페이지 완료

- ☒ DT-26842: [투어비스 공통] PC/MO 최근 본 여행 GA 수정 요청 건 완료

- 2024-04-19

- ☒ DT-26498: [투어비스] 오마카세 : PC 프론트 TA 로깅 작업 완료

- 2024-04-09

- ☒ DT-26231: [QA][PROD] 투어비스 PC > 최근 본 여행 : 최근 본 여행

- 스크롤 전위 오류 완료

- ☒ DT-26303: [투어비스] 모바일 하단 탭바 메뉴 추가 완료

- 2024-04-08

- ☒ DT-21499: [투어비스] 공통 컴포넌트 개발 완료

- 2024-02-29

- 숙세페 조기 종료 일정 적용함. 2024-02-29 03-03 06:59:59