

??-????

????????????

1. ????

1.1 ???

| ?? | ?? | ?? | ?? |
|---------|------------------------|--------|---------------------------|
| ??? | Vue.js | 3.x | ?? Composition API ??? |
| UI??? | Element Plus | ??? | ???? UI??? |
| ???? | ECharts | 5.x | ???????? |
| HTTP??? | Axios | 0.27.2 | ???????? API?? |
| ??? | Vuex | 4.x | ???????? |
| ??? | Spring Boot | 2.5.x | ?? RESTful API?? |
| ???? | MyBatis Plus | 3.5.x | ???????? |
| ??? | MySQL | 5.7+ | ???????? |
| API?? | Amazon Advertising API | ??? | ???????? |

1.2 ????

????????????????????

- ?? ?
 - ?? Vue????????
 - ???? Vue?? API????
 - API???? API????
- ?? ?
 - ?? Spring MVC???? HTTP??
 - ????????
 - ???? MyBatis Plus Mapper????
 - ?? API?? Amazon Advertising API????

3. 测试 1

- 测试用例
- 测试数据

1.3 测试

sequenceDiagram

```
participant User as 用户
participant Frontend as 前端
participant API as API
participant Backend as 后端
participant Service as 服务
participant Mapper as Mapper
participant DB as DB
participant AmazonAPI as AmazonAPI
```

User->>Frontend: 请求数据

Frontend->>API: 请求数据

API->>Backend: GET /api/v1/advSummary

Backend->>Service: 调用getenablesumtype()

Service->>Mapper: 调用getenablesumtype()

Mapper->>DB: SELECT * FROM ad_campaigns WHERE status='enabled'

DB-->>Mapper: 返回数据

Mapper-->>Service: 返回数据

Service-->>Backend: 返回数据

Backend-->>API: 200 OK { campaigns: 10, adGroups: 20, ... }

API-->>Frontend: 返回数据

Frontend->>API: 请求数据

API->>Backend: GET /api/v1/advSummary/warning

Backend->>Service: 调用getProductWarningIndicator()

Service->>Mapper: 调用getProductWarningIndicator()

Mapper->>DB: SELECT * FROM ad_warning WHERE type='productads'

DB-->>Mapper: 返回数据

Mapper-->>Service: 返回数据

Service-->>Backend: 返回数据

Backend-->>API: 200 OK { impco: 5, clickco: 3, ... }

API-->>Frontend: 返回数据

```

Frontend->>API: []
API->>Backend: POST /api/v1/advReport/getsumproduct
Backend->>Service: []getSumProduct()
Service->>Mapper: []
Mapper->>DB: SELECT * FROM ad_summary WHERE date BETWEEN ? AND ?
DB-->>Mapper: []
Mapper-->>Service: []
Service-->>Backend: []
Backend-->>API: 200 OK { summary: {...}, chartdata: {...} }
API-->>Frontend: []

```

2. ????

2.1 ????

| UI | UI | UI | UI |
|---------|---|----|--|
| UI | wimoor-ui/src/views/amazon/advertisement/overview/index.vue | UI | loadWaringData(), loadWaringDataDetail() |
| UI | wimoor-ui/src/views/amazon/advertisement/overview/components/adStatistics.vue | UI | loadSummaryChartData(), loadMonthSummaryData(), refreshChart() |
| UI | wimoor-ui/src/views/amazon/advertisement/overview/components/adFunnel.vue | UI | - |
| ROAS UI | wimoor-ui/src/views/amazon/advertisement/overview/components/roasRank.vue | UI | - |
| UI | wimoor-ui/src/views/amazon/advertisement/overview/components/indicator_detail.vue | UI | - |
| UI | wimoor-ui/src/views/amazon/advertisement/overview/components/indicator.vue | UI | - |

2.2 ???????

2.2.1 ???????


```

    })
}

// 加载月总结数据
function loadMonthSummaryData() {
    summaryApi.getmonthsum(state.queryParams).then(res=>{
        state.monthData=res.data;
        if(state.queryParams.currency=="USD"){
            state.rows.forEach(item=>{
                if(item.prefix==''){
                    item.prefix='$';
                }
            })
        }else{
            state.rows.forEach(item=>{
                if(item.prefix=='$'){
                    item.prefix='';
                }
            })
        }
    });
}

// 刷新图表
function refreshChart() {
    var chartdata=state.chartData;
    if (chartdata != null) {
        var labels = null;
        var color = [];
        var legends = [];
        var series = [];
        state.adList.forEach(row=>{
            var type = row.field;
            if (row.active&&chartdata[type]) {
                labels = chartdata[type]["listLabel"];
                legends.push(row.name);
                color.push(row.color);
                var datas = {};
                datas.name = row.name;
                if (type == "cr" || type == "ctr" || type == "acos") {

```

```

        datas.type = "line";
        datas.yAxisIndex = 1;
        datas.symbol = 'emptycircle';
        datas.smooth = true;
        datas.symbolSize = 3;
        datas.itemStyle = {
            normal : {
               LineStyle : {
                    width : 2
                }
            }
        }
    } else {
        datas.type = "bar";
        datas.barGap = "0%";
        datas.boundaryGap = "0%";
        datas.barMaxWidth = 32, datas.itemStyle = {
            normal : {
                barBorderRadius : [ 4, 4, 0, 0 ]
            }
        }
    }
    datas.data = chartdata[type].listData;
    series.push(datas);
}
});
if (labels != null) {
    lineChart(legends, labels, series, color);
} else {
    document.getElementById("mychart").innerHTML="<div style='padding-top:10%'
clas='font-extraSmall'>□□□□</div>";
}
} else {
    document.getElementById("mychart").innerHTML="<div style='padding-top:10%' clas='font-
extraSmall'>□□□□</div>";
}
}
}

```

2.2.4 ?????

□□□□ □

- [] API []
- [] ECharts []
- []

2.2.5 ROAS??

[] []

- [] API [] ROAS []
- [] ROAS []
- []

2.3 API??

| API [] | [] | URL | [] | [] | [] |
|-----------------------------|------|---------------------------------|------------|---|--|
| getenablesumtype | GET | /api/v1/advSummary | [] [] | [] | { campaigns: 10, adGroups: 20, ads: 30, targets: 40 } |
| getProductWarningIndicator | GET | /api/v1/advSummary/warning | [] [] | ftype: "co"/"sequent"/"yesterday" | { impco: 5, clickco: 3, crco: 2, acosco: 4 } |
| getKeywordsWarningIndicator | GET | /api/v1/advSummary/warning | [] [] | ftype: "co"/"sequent"/"yesterday" | { impco: 2, clickco: 1, crco: 0, acosco: 3 } |
| getsumproduct | POST | /api/v1/advReport/getsumproduct | [] [] | begin: "2023-01-01", end: "2023-01-31", groupid: "1", profileid: "2", currency: "USD" | { summary: {...}, chartdata: {...}, ordersummary: 1000 } |
| getmonthsum | POST | /api/v1/advReport/getmonthsum | [] [] | begin: "2023-01", end: "2023-03", groupid: "1", profileid: "2", currency: "USD" | { impressions: {...}, clicks: {...}, ... } |

3. ????

3.1 ???

| | | | |
|-----|-----|-----|-----|
| [] | [] | [] | [] |
|-----|-----|-----|-----|

| | | | |
|-------------------------|---|-------|---|
| AdvertReportController | wimoor-amazon-adv/amazon-adv-boot/src/main/java/com/wimoor/amazon/adv/controller/AdvertReportController.java | □□□□□ | getSumProductAction(), getMonthSumAction() |
| AdvertManagerController | wimoor-amazon-adv/amazon-adv-boot/src/main/java/com/wimoor/amazon/adv/controller/AdvertManagerController.java | □□□□□ | - |

3.2 ????

| □□□ | □□□ | □□□ | □□□ |
|-------------------------------|--|----------|--|
| AmzAdvSumServiceImpl | wimoor-amazon-adv/amazon-adv-boot/src/main/java/com/wimoor/amazon/adv/service/impl/AmzAdvSumServiceImpl.java | □□□□□□□ | - |
| AmzAdvSumProductAdsService | wimoor-amazon-adv/amazon-adv-boot/src/main/java/com/wimoor/amazon/adv/report/service/IAmzAdvSumProductAdsService.java | □□□□□□□□ | getSumProduct(), getMonthSumProduct(), getDaysSumProduct() |
| AmazonReportAdvSummaryService | wimoor-amazon-adv/amazon-adv-boot/src/main/java/com/wimoor/amazon/adv/common/service/IAmazonReportAdvSummaryService.java | □□□□□□□ | findAdvert() |

3.3 ?????

| □□□ | □□□ | □□□ | □□□ |
|------------------------|-------|---------|---|
| AmzAdvSumProductAds | □□□□□ | □□□□□□□ | campaignId, adGroupId, adId, impressions, clicks, cost, attributedSales, attributedUnitsOrdered |
| AmzAdvWarning | □□□□□ | □□□□□□□ | type, subtype, indicator, value, threshold, status |
| AmazonReportAdvSummary | □□□□□ | □□□□□□□ | date, profileId, campaignId, adGroupId, adId, impressions, clicks, cost, sales |

3.4 ?????

| Mapper | 接口 | 实现类 | 方法 |
|------------------------------|-------------------------------|-----------------------------------|---|
| AmzAdvSumProductAdsMapper | AmzAdvSumProductAdsService | AmzAdvSumProductAdsServiceImpl | selectSumProduct(), selectMonthSumProduct(), selectDaysSumProduct() |
| AmzAdvWarningMapper | AmzAdvWarningService | AmzAdvWarningServiceImpl | selectWarningIndicator() |
| AmazonReportAdvSummaryMapper | AmazonReportAdvSummaryService | AmazonReportAdvSummaryServiceImpl | selectAdvert() |

3.5 API

3.5.1 API

接口

- AmzAdvSumProductAdsService
- AmzAdvSumProductAdsService.getSumProduct() ID
- AmzAdvSumProductAdsService.orderSummaryAll()
- AmzAdvSumProductAdsService.getDaysSumProduct()
- AmzAdvWarningService

实现类

```

@PostMapping("/getsumproduct")
public Result<Map<String, Object>> getSumProductAction(@RequestBody QueryForSumProductDTO
dto){
    String begin =dto.getBegin();
    String end = dto.getEnd();
    String type = dto.getType();
    String groupid = dto.getGroupid();
    String profileid = dto.getProfileid();
    String currency = dto.getCurrency();
    UserInfo user = UserInfoContext.get();
    Map<String,Object> param=new HashMap<String,Object>();
    param.put("shopid", user.getCompanyid());
    param.put("type", type);
    param.put("currency", currency);
    Map<String, Marketplace> allmarket = marketplaceService.findMapByMarketplaceId();
    if(StringUtil.isNotEmpty(profileid) && !"all".equals(profileid)) {
        if(StringUtil.isNotEmpty(groupid) && !"all".equals(groupid)) {
            AmzAdvProfile profile = amzAdvAuthService.getAmzAdvProfileByKey(new
BigInteger(profileid));

```

```

        if(allmarket.get(profile.getMarketplaceid())!=null){
            param.put("pmarketplaceId",
allmarket.get(profile.getMarketplaceid()).getPointName());
        }
        param.put("sellerid", profile.getSellerid());
        param.put("profileid", profileid);
        param.put("groupid", groupid);
    }else {
        param.put("marketplaceId", profileid);
        param.put("pmarketplaceId", allmarket.get(profileid).getPointName());
    }
}else {
    if(StringUtil.isNotEmpty(groupid) && !"all".equals(groupid)) {
        List<Map<String, Object>> list = amzAdvAuthService.getSelleridBygroup(groupid);
        List<String> sellerList = new ArrayList<String>();
        for(Map<String,Object> map : list) {
            String seller = (String) map.get("sellerId");
            sellerList.add(seller);
        }
        param.put("sellerList", sellerList);
        param.put("groupid", groupid);
    }
}
if(StringUtil.isNotEmpty(begin)) {
    param.put("begin", begin.replaceAll("/", "-").trim());
    param.put("beginDate", begin.replaceAll("/", "-").trim());
}
if(StringUtil.isNotEmpty(end)) {
    param.put("end",end.replaceAll("/", "-").trim());
    param.put("endDate", end.replaceAll("/", "-").trim());
}
Map<String, Object> result =new HashMap<String,Object>();
Map<String, Object> map = amzAdvSumProductAdsService.getSumProduct(param);
BigDecimal mapordersum = amzAdvSumProductAdsService.orderSummaryAll(param);
Map<String, Object> chartdata = amzAdvSumProductAdsService.getDaysSumProduct(param);
result.put("summary", map);
result.put("ordersummary", mapordersum);
result.put("chartdata", chartdata);
return Result.success(result) ;
}

```


- []
- []

[] []

- [] API []
- [] SQL []
- [] ECharts []

[] []

- []
- []
- []

4.2 ???????

[] []

- []
- []
- []
- []

ACOS []

[] []

- []
- []
- []

[] []

- []
- []
- []

4.3 ???????

[] []

- []
- [] Excel []
- []

[] []

- 使用 Apache POI 和 Excel
- 使用 HTTP 和 Excel

使用

- 使用
- 使用
- 使用

4.4 ???????

使用

- 使用
- 使用
- 使用

使用

- 使用
- 使用 ECharts

使用

- 使用
- 使用
- 使用

4.5 ROAS????

使用

- 使用
- 使用
- 使用

使用

- 使用 ROAS
- 使用 ROAS

使用

- 使用
- 使用

□□□□ □

- □□ Spring Security□□□□□
- □□□□□□□□□□
- □□□□□□□

6.2 ?????

□□□□ □

- □□□□□□□□□□□□□□
- □□□□□□□□□□□□□□□□
- □□□□□□□□□□□□□□□□

□□□□ □

- □□ HTTPS□□□□□□
- □□□□□□□□
- □□□□□□□□□□

6.3 ????

□□□□ □

- □□□□□□□□ API□□□□
- □□□□□□□□□□
- □□□□□□□□□□□□

□□□□ □

- □□□□□□□□□□ API□□
- □□□□□□□□
- □□□□□□□□□□

7. ??????

7.1 ?????

□□□□ □

- □□□□□□□□□□ Sponsored Brands□ Sponsored Display□
- □□□□□□□□ CPC□□□ ROAS□□□□
- □□□□□□□□□□□□□□□□□□

- []

[] []

- []
- []
- []

7.2 ?????

[] []

- []
- []
- []
- [] API [] API

[] []

- []
- [] Docker []
- []

7.3 ?????

[] []

- []
- []
- []

[] []

- [] API [] API []
- []
- Webhooks [] Webhooks []

8. ???????

8.1 ?????

[] []

1. 组件 生命周期
2. 组件 生命周期 Vuex 生命周期 props
3. 组件 生命周期
 - 生命周期钩子
 - 生命周期钩子
 - 生命周期钩子
4. 组件 生命周期
 - 生命周期钩子 ESLint 生命周期钩子
 - 生命周期钩子
 - 生命周期钩子 Vue 生命周期钩子 computed 生命周期钩子

组件 生命周期

```
// 生命周期钩子
function handleQuery() {
  if(state.activeName=="chart"){
    loadSummaryChartData();
  }else{
    loadMonthSummaryData() ;
  }
}

// 生命周期钩子
import { debounce } from 'lodash-es';

const handleQuery = debounce(() => {
  if(state.activeName=="chart"){
    loadSummaryChartData();
  }else{
    loadMonthSummaryData() ;
  }
}, 300);
```

8.2 生命周期

组件 生命周期

1. 组件 生命周期
 - 生命周期钩子
 - 生命周期钩子 SQL 生命周期钩子
 - 生命周期钩子
2. 组件 生命周期

- Redis

-

-

3.

-

- Lambda Stream API

-

4.

-

-

```
// 
if(StringUtil.isEmpty(begin)) {
    param.put("begin", begin.replaceAll("/", "-").trim());
    param.put("beginDate", begin.replaceAll("/", "-").trim());
}
if(StringUtil.isEmpty(end)) {
    param.put("end",end.replaceAll("/", "-").trim());
    param.put("endDate", end.replaceAll("/", "-").trim());
}

// 
private void processDateParams(Map<String, Object> param, String begin, String end) {
    if(StringUtil.isEmpty(begin)) {
        String formattedBegin = begin.replaceAll("/", "-").trim();
        param.put("begin", formattedBegin);
        param.put("beginDate", formattedBegin);
    }
    if(StringUtil.isEmpty(end)) {
        String formattedEnd = end.replaceAll("/", "-").trim();
        param.put("end", formattedEnd);
        param.put("endDate", formattedEnd);
    }
}
}
```

8.3 ???????

- 1.

2.
3.
4.

```

// 测试接口
@PostMapping("/getsumproduct")
public Result<Map<String, Object>> getSumProductAction(@RequestBody QueryForSumProductDTO dto)
{
    // 测试接口
    Map<String, Object> map = amzAdvSumProductAdsService.getSumProduct(param);
    BigDecimal mapordersum = amzAdvSumProductAdsService.orderSummaryAll(param);
    Map<String, Object> chartdata = amzAdvSumProductAdsService.getDaysSumProduct(param);
    // ...
}

// 测试接口
@PostMapping("/getsumproduct")
public Result<Map<String, Object>> getSumProductAction(@RequestBody QueryForSumProductDTO dto)
{
    // 测试接口
    Map<String, Object> precomputedData =
amzAdvSumProductAdsService.getPrecomputedSumProduct(param);
    if (precomputedData != null) {
        return Result.success(precomputedData);
    }
    // 测试接口
    Map<String, Object> map = amzAdvSumProductAdsService.getSumProduct(param);
    BigDecimal mapordersum = amzAdvSumProductAdsService.orderSummaryAll(param);
    Map<String, Object> chartdata = amzAdvSumProductAdsService.getDaysSumProduct(param);
    // 测试接口
    amzAdvSumProductAdsService.cacheSumProductResult(param, map, mapordersum, chartdata);
    // ...
}

```

9. ??

9.1 ?????

