

1、添加指定时间点截图任务

描述

通过视频id、截图等相关参数添加视频指定时间点截图任务
接口地址（仅做说明使用）：<https://api.polyv.net/v2/video/snapshot/%s/addTask>

调用约束

- 1、接口调用有频率限制，[详细请查看](#)，调用常见异常，[详细请查看](#)
- 2、接口用于添加时间点截图任务，每个任务允许截图20张
- 3、禁播的视频不允许截图操作
- 4、如设置了callbackUrl，值为<http://example.polyv.net/snapshot-callback.do> 那么截图任务完成后，polyv会回调该接口，

并带上签名信息，开发者可以通过签名信息来校验调用是否为polyv的合法调用，具体的签名规则：
`md5("snapshot" + vid + secretKey)`。

如vid="e6b23c6f51350f106556806a576b1942_e"，secretKey="testKey"，那么
sign="3adb60893894d422d00ed2efae8c41f3"

(小写md5)。最终回调的url为<http://example.polyv.net/snapshot-callback.do?sign=3adb60893894d422d00ed2efae8c41f3>

单元测试

```
@Test
public void testCreateScreenshotTask() throws IOException, NoSuchAlgorithmException {
    VodCreateScreenshotTaskRequest vodCreateScreenshotTaskRequest = new VodCreateScreenshotTaskRequest();
    Integer vodCreateScreenshotTaskResponse = null;
    try {
        vodCreateScreenshotTaskRequest.setUploadTime(new Date())
            .setVideoId(super.getTestVideoId())
            .setOffsetTimes("8");
        vodCreateScreenshotTaskResponse = new VodScreenshotServiceImpl().createScreenshotTask(
            vodCreateScreenshotTaskRequest);
        Assert.assertNotNull(vodCreateScreenshotTaskResponse);
        if (vodCreateScreenshotTaskResponse != null) {
            log.debug("测试添加指定时间点截图任务成功,{}", vodCreateScreenshotTaskResponse);
        }
    } catch (PloyvSdkException e) {
        //参数校验不合格 或者 请求服务器端500错误, 错误信息见PloyvSdkException.getMessage()
        log.error(e.getMessage(), e);
        // 异常返回做B端异常的业务逻辑, 记录log 或者 上报到ETL 或者回滚事务
        throw e;
    } catch (Exception e) {
        log.error("SDK调用异常", e);
        throw e;
    }
}
```

单元测试说明

- 1、请求正确，返回Integer对象，B端依据此对象处理业务逻辑；
- 2、请求参数校验不合格，抛出PloyvSdkException，错误信息见PloyvSdkException.getMessage()，如 [输入参数 [xxx.chat.VodxxxRequest]对象校验失败，失败字段 [pic不能为空 / msg不能为空]]
- 3、服务器处理异常，抛出PloyvSdkException，错误信息见PloyvSdkException.getMessage()，如 [保利威请求返回数据错误，请求流水号：66e7ad29fd04425a84c2b2b562d2025b，错误原因： invalid signature.]

请求入参描述

参数名	必选	类型	说明
uploadTime	true	Date	上传时间，格式： yyyy-MM-dd HH:mm:ss 【对应api文档的ptime字段】

参数名	必选	类型	说明
videoid	true	String	视频ID【对应api文档的vid字段】
offsetTimes	true	String	截图时间点，多个时间点以英文逗号分隔，单位：秒
width	false	Integer	截图的宽度，默认为原视频宽
height	false	Integer	截图的高度，默认为原视频高
callbackUrl	false	String	截图完成后的回调地址，截图任务完成会post结果信息和签名到回调的地址，若地址返回的Http状态码为200，则会视为回调成功。例如 http://example.polyv.net/snapshot-callback.do

返回对象描述

添加成功返回截图任务id

2、查询截图任务状态

描述

通过截图任务id查询截图任务状态 接口地址（仅做说明使用）： https://api.polyv.net/v2/video/snapshot/%s/getTaskStatus

调用约束

1、接口调用有频率限制，[详细请查看](#)，调用常见异常，[详细请查看](#)

单元测试

```
@Test
public void testGetScreenshotTaskStatus() throws IOException, NoSuchAlgorithmException {
    VodGetScreenshotTaskStatusRequest vodGetScreenshotTaskStatusRequest = new
VodGetScreenshotTaskStatusRequest();
    VodGetScreenshotTaskStatusResponse vodGetScreenshotTaskStatusResponse = null;
    try {
        vodGetScreenshotTaskStatusRequest.setTaskId(1146);
        vodGetScreenshotTaskStatusResponse = new VodScreenshotServiceImpl().getScreenshotTaskStatus(
            vodGetScreenshotTaskStatusRequest);
        Assert.assertNotNull(vodGetScreenshotTaskStatusResponse);
        if (vodGetScreenshotTaskStatusResponse != null) {
            log.debug("测试查询截图任务状态成功,{})", JSON.toJSONString(vodGetScreenshotTaskStatusResponse));
        }
    } catch (PloyvSdkException e) {
        //参数校验不合格 或者 请求服务器端500错误, 错误信息见PloyvSdkException.getMessage()
        log.error(e.getMessage(), e);
        // 异常返回做B端异常的业务逻辑, 记录log 或者 上报到ETL 或者回滚事务
        throw e;
    } catch (Exception e) {
        log.error("SDK调用异常", e);
        throw e;
    }
}
```

单元测试说明

- 1、请求正确，返回VodGetScreenshotTaskStatusResponse对象，B端依据此对象处理业务逻辑；
- 2、请求参数校验不合格，抛出PloyvSdkException，错误信息见PloyvSdkException.getMessage()，如 [输入参数 [xxx.chat.VodxxxRequest]对象校验失败，失败字段 [pic不能为空 / msg不能为空]]
- 3、服务器处理异常，抛出PloyvSdkException，错误信息见PloyvSdkException.getMessage()，如 [保利威请求返回数据错误，请求流水号：66e7ad29fd04425a84c2b2b562d2025b，错误原因： invalid signature.]

请求入参描述

参数名	必选	类型	说明
taskId	true	Integer	任务ID

返回对象描述

参数名	类型	说明
taskId	Integer	任务id
vid	String	视频vid
status	String	任务状态，waiting-等待截图,processing-截图处理中,success-任务成功,fail-任务失败
createTime	Date	任务创建时间,时间格式 yyyy-MM-dd HH:mm:ss
beginProcessTime	Date	开始截图的时间，时间格式 yyyy-MM-dd HH:mm:ss
finishProcessTime	Date	完成截图的时间，时间格式 yyyy-MM-dd HH:mm:ss
screenshots	Array	截图信息组，失败或未开始时为空【对应api文档的 snapshots 字段】 【详见 Screenshot参数描述 】

Screenshot参数描述

参数名	类型	说明
offsetTime	Integer	截图时间点，单位：秒
imageUrl	String	截图访问的url