I. Các API, hàm, cấu trúc hỗ trợ

1. Cấu trúc class ResponseExceptionMessage

```
public class ResponseExceptionMessage
{
    public string Message { get; set; }
    public string ExceptionMessage { get; set; }
    public string ExceptionType { get; set; }
    public string StackTrace { get; set; }
}
```

- Tất cả api khi thực thi không thành công sẽ trả về chuỗi có thể parse về cấu trúc

ResponseExceptionMessage.

2. Các hàm hỗ trợ

```
private static string Compress(string text)
    byte[] buffer = Encoding.UTF8.GetBytes(text);
    MemoryStream ms = new MemoryStream();
    using (GZipStream zip = new GZipStream(ms, CompressionMode.Compress, true))
        zip.Write(buffer, 0, buffer.Length);
    }
    ms.Position = 0;
   MemoryStream outStream = new MemoryStream();
    byte[] compressed = new byte[ms.Length];
    ms.Read(compressed, 0, compressed.Length);
    byte[] gzBuffer = new byte[compressed.Length + 4];
    System.Buffer.BlockCopy(compressed, 0, gzBuffer, 4, compressed.Length);
    System.Buffer.BlockCopy(BitConverter.GetBytes(buffer.Length), 0, gzBuffer, 0, 4);
    return Convert.ToBase64String(gzBuffer);
private static string Decompress(string compressedText)
    byte[] gzBuffer = Convert.FromBase64String(compressedText);
    using (MemoryStream ms = new MemoryStream())
        int msgLength = BitConverter.ToInt32(gzBuffer, 0);
        ms.Write(gzBuffer, 4, gzBuffer.Length - 4);
        byte[] buffer = new byte[msgLength];
        ms.Position = 0;
        using (GZipStream zip = new GZipStream(ms, CompressionMode.Decompress))
            zip.Read(buffer, 0, buffer.Length);
        return Encoding.UTF8.GetString(buffer);
    }
}
public static string DataSetToXMLString(DataSet dts)
    string str = Compress(dts.GetXml());
    return str;
}
public static DataSet DataSetFromXMLString(string xml)
    string strContent = Decompress(xml);
    DataSet dts = new DataSet();
    dts.ReadXml(new System.IO.StringReader(strContent), XmlReadMode.Auto);
}
```

3. Lấy danh mục

Link: api/danhmuc/?type={loaidanhmuc}. Với {loaidanhmuc} gồm có:

- DanToc
- Huyen
- Xa
- KhuyetTat
- TonGiao
- DoiTuongChinhSach
- DotBaoCao
- PhongGD
- BienDong
- ThongTinTruong
- DanhMucTruong
- Lop
- Lop PrintMode
- HocSinh
- HocSinh PrintMode
- HocSinh DangHoc PrintMode
- HocSinh_ChuyenDen_PrintMode
- HocSinh ChuyenDi PrintMode
- HocSinh_ThoiHoc_PrintMode

Đúng: trả về Dataset danh mục

Sai: Json Exception

II. Quy trình nộp số liệu đầu năm.

1. Yêu cầu một phiên làm việc

Link:

api/getToken/{systemUsing}/{password}/{schoolID}/{clientUserID}/{clientIP}

Đúng: trả về chuỗi **token**

Sai: Json Exception

2. Get captcha

Link: Home/getCaptcha/{token}

Đúng: Trả về một ảnh captcha (file stream)

Sai: Trả về chuỗi lỗi.

3. Submit captcha

Link: api/submitCaptcha/{token}/{captchaCode}

Đúng: Trả về chuỗi "verified"

Sai: Trả về chuỗi lỗi.

Sau khi yêu cầu một phiên làm việc thành công (sau khi thành công bước 3). Mọi api khi gởi yêu cầu chỉ cần truyền một header "**TokenID**" với giá trị của TokenID bạn đã được cấp ở bước 1.

4. Nộp học sinh (Phương thức post).

Link: api/hocsinh/?dotbaocaoid={0}

Post Content: DataSetToXMLString(dtsHocSinh) với content type = "text/plain"; Đúng: Trả về một dataset trong đó có một table có tên "result" chứa thông báo nộp thành công.

Sai: Trả về một dataset trong đó có một table có tên "error" chứa tất cả những dòng dữ liệu bị lỗi.

Cấu trúc dtsHocSinh (dataset có 1 table duy nhất có cấu trúc)		
HocSinhID	String (max = 50)	Mã học sinh do Sở cấp - Trường hợp nộp học sinh mới
		thì để trống HocSinhID
		- Trường hợp cập nhật thông
		tin học sinh Bắt buộc phải có
		HocSinhID
ClientHocSinhID	String (max = 50)	Mã học sinh của từng hệ thống quản lý
		- Khi nộp học sinh mới, bắt
		buộc phải có ClientHocSinhID
Khoi	Int	Khối phải năm trong phạm vi 1 đến 12
TenLop		den 12
LoaiBienDongID	Int	Mã biến động đã đồng bộ với
		danh mục biến động của sở. Vui
		lòng lấy từ API với loại danh
		mục là "BienDong").
NgayBienDong		,
LyDoBienDong		Ghi chú của loại biến động
HuyenID		Mã huyện đã đồng bộ với danh
		mục huyện của sở. Vui lòng lấy
		từ API với loại danh mục là
		"Huyen").
		Nếu học sinh là học sinh chuyển
		đến từ trường khác. Bắt buộc
T		phải có HuyenID
Truong		Tên trường chuyển đi hoặc
		chuyển đến. Nếu huyện nằm
		trong khu vực thành phố thì
		Trường = Mã trường, còn từ
		tỉnh khác chuyển đến thì
G O (D' 1		Trường = Tên trường
SoQuyetDinh		Số quyết định chuyển đến. Bắt
		buộc phải có khi học sinh ngoài
		tỉnh chuyển đến.

Но	String ($max = 50$)	
Ten	String (max = 20)	
GioiTinh	Bool	Giới tính (true = nữ; false =
		nam)
NgaySinh	String $(max = 20)$	Ngày sinh (Định dạng
		dd/MM/yyyy)
NoiSinh	String (max = 100)	
DanTocID	Int	Mã dân tộc đã đồng bộ với danh
		mục dân tộc của sở. (Vui lòng
		lấy từ (API với loại danh mục là
		"DanToc").
TonGiaoID	Int	Mã tôn giáo đã đồng bộ với
		danh mục tôn giáo của sở. Vui
		lòng lấy từ API với loại danh
***	-	mục là "TonGiao").
KhuyetTatID	Int	Mã khuyết tật đã đồng bộ với
		danh mục khuyết tật của sở.
		(Vui lòng lấy từ API với loại
D 'T C1' 10 1ID	т ,	danh mục là "KhuyetTat").
DoiTuongChinhSachID	Int	Mã đổi tượng chính sách đã
		đồng bộ với danh mục đối tượng chính sách của sở. (Vui
		lòng lấy từ API với loại danh
		mục là "DoiTuongChinhSach").
HoKhau DiaChi	String (max = 500)	Số nhà theo hộ khẩu
HoKhau XaID	int	Xã theo hộ khẩu
DCTT DiaChi	String (max = 500)	Số nhà theo địa chỉ thương trú
DCTT_Dlacin	int	Xã theo Hộ khẩu thường trú
SDT	String (max = 50)	The theo Ho khaa thaong tha
TenCha	String (max $= 50$)	
SDTCha	String (max $= 50$)	
TenMe	String (max $= 50$)	
SDTMe	String (max = 50)	
TenNguoiGiamHo	String (max = 50)	
SDTNguoiGiamHo	String $(max = 50)$	
CMND	String $(max = 50)$	
KÉT QUẢ	0	học sinh đã nộp gọi API với loại danh
	mục là "HocSinh"	

5. Nộp lớp.

Link: api/lop

Post Content: DataSetToXMLString(dtsLopHoc) với content type = "text/plain";

Đúng: Trả về một dataset trong đó có một table có tên "result" chứa thông báo nộp

thành công.

Sai: Trả về một dataset trong đó có một table có tên "error" chứa tất cả những dòng dữ liệu bị lỗi.

Cấu trúc dtsLopHoc (dataset có 1 table duy nhất có cấu trúc)			
LopID	String ($max = 50$)	Mã lớp do Sở cấp	
STT	Int	Thứ tự lớp	
Khoi	Int	Khối phải nằm trong phạm vi 1	
		đến 12	
TenLop	String (max = 50)		
PGDID	Int	Mã PGD đã đồng bộ với danh mục PGD của sở. Vui lòng lấy từ API với loại danh mục là "PhongGD").	
Hoc2BuoiNgay	Bool		
	Để lấy lại danh sách lớp đã nộp gọi API với loại danh mục là "lop"		

6. Nộp biến động học sinh (Phương thức post).

Link: api/hocsinhbiendong/?dotbaocaoid={0}

Post Content: DataSetToXMLString(dtsHocSinh) với content type = "text/plain";

Đúng: Trả về một dataset trong đó có một table có tên "result" chứa thông báo nộp

thành công.

Sai: Trả về một dataset trong đó có một table có tên "error" chứa tất cả những dòng dữ liệu bị lỗi.

Cấu trúc dtsHocSin	Cấu trúc dtsHocSinh (dataset có 1 table duy nhất có cấu trúc) Ghi chú			
HocSinhID	String (max	Mã học sinh do Sở cấp	Bắt	
	= 50)	- Trường hợp nộp học sinh mới thì để	buộc*	
	,	trống HocSinhID		
		- Trường hợp cập nhật thông tin học		
		sinh Bắt buộc phải có HocSinhID		
TenLop			Bắt buộc	
LoaiBienDongID	Int	Mã biến động đã đồng bộ với danh mục	Bắt buộc	
		biến động của sở. Vui lòng lấy từ API		
		với loại danh mục là "BienDong").		
NgayBienDong			Bắt buộc	
LyDoBienDong		Ghi chú của loại biến động		
HuyenID		Mã huyện đã đồng bộ với danh mục		
		huyện của sở. Vui lòng lấy từ API với		
		loại danh mục là "Huyen").		
		Nếu học sinh là học sinh chuyển đến từ		
		trường khác. Bắt buộc phải có HuyenID		
Truong		Tên trường chuyển đi hoặc chuyển đến.		
		Nếu huyện nằm trong khu vực thành phố		
		thì Trường = Mã trường, còn từ tỉnh		
		khác chuyển đến thì Trường = Tên		
		trường		

SoQuyetDinh	Số quyết định chuyển đến. Bắt buộc	
	phải có khi học sinh ngoài tỉnh chuyển	
	đến.	

Field HocSinhID: Bắt buộc cho tất cả các biến động, ngoại trừ biến động Chuyển đến từ ngoài tỉnh.

Biến động chuyển đến:

- Field HuyenID: Từ huyện nào đến, bắt buộc
- Field SoQuyetDinh: (Số quyết định chuyển đi), bắt buộc
- Chuyển đến trong tỉnh:
 - o Field HocSinhID: Bắt buộc
 - Field Truong: SchoolID của trường cũ (mã trên CSDL dùng chung của Sở), bắt buôc
 - Cưu ý: Trường cũ của học sinh phải nộp học sinh chuyển đi trước khi trường mới nộp học sinh chuyển đến. Học sinh chuyển đến sẽ được chuyển sang trường mới tức thì.
- Ngoài tỉnh
 - Field HocSinhID: Để trống
 - o Field Truong: Tên trường cũ, Bắt buộc
 - o Lưu ý: Học sinh chuyển đến sẽ ở trạng thái chờ duyệt.

Biến động chuyển đi:

- Field HocSinhID: Bắt buộc
- Field HuyenID: Huyện chuyển đến (đến huyện nào), Bắt buộc
- Field SoQuyetDinh: (Số quyết tiếp nhận), bắt buộc
- Chuyển đi trong tỉnh:
 - Field Truong: Mã trường mới SchoolID (mã trên CSDL dùng chung của Sở), bắt buộc.
 - Lưu ý: Học sinh chuyển đi sẽ được ghi nhận rời trường và trường cũ không còn thao tác được trên học sinh này nữa.
- Ngoài tỉnh
 - o Field Truong: Tên trường mới, Bắt buộc.
 - Lưu ý: Học sinh chuyển đi ở trạng thái chờ duyệt. Khi được duyệt, học sinh sẽ được ghi nhận là đã rời trường cũ, và trường cũ không còn thao tác được trên học sinh này nữa.

III. Quy trình đăng ký kỳ thi chung

- 1. Kết quả trả về của các lệnh gọi API
- Json object:

```
public class CallResult
{
    public int ErrorCode { get; set; }
    public String ErrorMessage { get; set; }
    public object Data { get; set; }
}
```

- Encrypted XML String: Descrypt và Deserialize như phần nộp dữ liệu đầu năm. Phân biệt 2 loại dữ liệu trả về bằng ký tự đầu tiên của chuổi: Có "{" hay không.

2. Các lệnh gọi

a. Get danh mục kỳ thi

api/KyThi/DanhMucKyThi

b. Get danh mục môn thi

api/KyThi/DanhMucMonThi/{kyThiID}

c. Submit thí sinh

Link: api/KyThi/ThiSinh/{kyThiID}/{monThiID}

Cấu trúc Dataset (dataset có 1 table duy nhất có cấu trúc)		
HocSinhID String (max = 50)		Mã học sinh do Sở cấp
Xoa	Boolean	Xoá học sinh đăng ký

d. Get thí sinh đã nộp

api/KyThi/GetThiSinh/{kyThiID}
api/KyThi/GetThiSinh/{kyThiID}/{monThiID}

6. Nộp điểm tổng kết cuối kỳ và cuối năm (Phương thức post).

```
Link: api/diemtk/{DotBaoCaoID}

Post Content: DataSetToXMLString(dtDiemTK) với content type = "text/plain";

Response:

Dataset

{

- Table1: {ErrorCode: Number, ErrorMessage: String}

• Row 0:

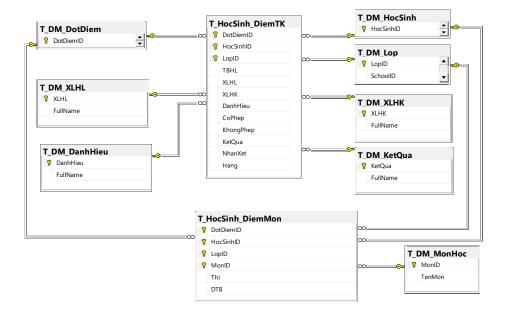
• ErrorCode:

• 0: Thành công

• != 0: Lỗi

- Table2 (option): Chi tiết lỗi của mỗi record nộp lên
}
```

Cấu trúc dtDiemTK (dataset có 1 table duy nhất có cấu trúc)		
HocSinhID	String	Mã học sinh do Sở cấp
TenLop	String	*Tên lớp, không phải mã lớp*
TBHL	Number	
XLHL	String	api/danhmuc/?type=XLHL
XLHK	String	api/danhmuc/?type=XLHK
DanhHieu	String	api/danhmuc/?type=DanhHieu
KetQua	String	api/danhmuc/?type=KetQua
CoPhep	Number	
KhongPhep	Number	
NhanXet	String	
Hang	Number	



7. Nộp điểm tổng kết và điểm thi cuối kỳ của từng môn học (Phương thức post).

```
Link: api/diemmon/{DotBaoCaoID}

Post Content: DataSetToXMLString(dtDiemMon) với content type = "text/plain";

Response:

Dataset

{

- Table1: {ErrorCode: Number, ErrorMessage: String}

• Row 0:

• ErrorCode:

• 0: Thành công

• != 0: Lỗi

- Table2 (option): Chi tiết lỗi của mỗi record nộp lên
```

Cấu trúc dtDiemTK (dataset có 1 table duy nhất có cấu trúc)		
HocSinhID	String	Mã học sinh do Sở cấp
TenLop	String	*Tên lớp, không phải mã lớp*
MonID	Number	api/danhmuc/?type=MonHoc
Thi	Number	
DTB	String	api/danhmuc/?type=XLHL