

گروه نرم افزاری تمشک

تحلیلگران ایده طبرستان

شماره ثبت ۱۱۵۲۹

گروه نرم افزاری تمشک

ارائه دهنده خدمات پیام کوتاه

راهنمای استفاده از سرویس

SOAP (Web service)

دفتر مرکزی : ساری – بلوار امیر مازندرانی – روبروی خیابان مهدی آباد – مجتمع زمرد طبقه چهارم

Website : Tameshksms.com

Email : info@tameshksms.com

Tel : ۰۱۱۲۳۳۹۴۸۵۴

sms center : ۰۵۰۰۰۵۷۰۵۰

فهرست مطالب ارائه شده در این راهنما :

۳	نگاه کلی	▲
۳	مقدمه	▲
۳	شروع به کار	▲
۴	گام اول - ایجاد یک حساب کاربری در سامانه پیامک تمشک	
۴	گام دوم - شروع به کار با سرویس SOAP	
۵	متد ها و پارامترها	▲
۵	SendSms	متد - ۱-۱
۹	GetCredit	متد - ۲-۱
۱۰	GetDelivery	متد - ۳-۱
۱۱	GetReceiveSmsWithNumber	متد - ۴-۱
۱۳	GetReceiveSms	متد - ۵-۱
۱۴	GetCountAll	متد - ۶-۱
۱۵	GetCount	متد - ۷-۱
۱۶	SendFutureSms	متد - ۸-۱
۱۸	CheckSendFuture	متد - ۹-۱

▲ توضیح موارد خطا

◆ نگاه کلی

مستند حاضر برای استفاده آن دسته از برنامه نویسانی طراحی شده که می خواهند با استفاده از نرم افزارهای آماده خود بر روی بستر SOAP سامانه پیامک تمشک ، به ارسال و دریافت پیامک بپردازند. در این راهنما روش ها و دستورات مورد نیاز برنامه نویسان در هنگام استفاده از سرویس SOAP شرح داده شده است.

این سرویس با توجه به آرایه ای بودن امکان ارسال همزمان چندین پیامک را در هر بار فراخوانی فراهم می سازد .

برای استفاده از سیستم SOAP شما باید ابتدا از طریق عقد قرارداد با سامانه پیامک تمشک حساب خود را فعال نمایید . پس از فعال سازی ، شما یک نام کاربری (USERNAME) ، کلمه عبور (PASSWORD) و نام دامنه (DOMAIN) اختصاصی دریافت می نمایید که برای استفاده در نرم افزار کاربر مورد استفاده قرار می گیرد .

بی تردید حفظ و نگهداری از این اطلاعات ضامن امنیت حساب کاربران خواهد بود .

در ادامه این متن ، استفاده از SOAP API را مورد بررسی قرار می دهیم .

◆ مقدمه

این روش از روشهای پیشرفته در انتقال پیامک به شمار می رود این متد ارسال و دریافت پیامک های فارسی ، لاتین و باینری با اندازه های متفاوت را دارا می باشد . برای برقراری ارتباط با API سامانه پیامک تمشک می توانید از دو پروتکل HTTP بر روی پورت ۸۰ ، و یا استفاده از پروتکل امن HTTPS بر روی پورت ۴۴۳ استفاده نمایید .

توجه : لطفا قبل از تماس با واحد پشتیبانی حتما این مستند را به صورت کامل مطالعه بفرمایید .

◆ شروع کار

برای استفاده از خدمات ارسال پیامک سامانه پیامک تمشک به یک حساب کاربری در پنل ارسال پیامک سامانه پیامک تمشک و حداقل یک مسیر ارتباطی مشخص (مثل اینترنت) بین نرم افزار طراحی شده خود و سرور پیام دهی سامانه پیامک تمشک نیاز خواهید داشت .

گام اول – ایجاد یک حساب کاربری در سامانه پیامک تمشک

در صورتیکه شما در حال حاضر دارای یک حساب کاربری در سامانه پیامک تمشک هستید از این مرحله عبور کرده و به گام دوم بروید . در صورتیکه شما در حال حاضر فاقد حساب کاربری هستید ، می توانید برای گشایش حساب از طریق وب سایت سیستم های سیستم های ارتباطی تماس حاصل فرمایید و از شرایط و مدارک لازم برای افتتاح حساب کاربری اطلاع حاصل نمایید (آدرس ایمیل ، شماره تلفن های تماس و فکس واحد فروش پیامک سامانه پیامک تمشک در قسمت تماس با ما درج شده است

<http://tameshksms.com>

پس از فعال شدن حساب کاربری ، شما یک نام کاربری (username) و کلمه عبور (password) و نام دامنه (domain) و یک یا چند شماره خواهید داشت .

هم اکنون با در اختیار داشتن اطلاعات ذکر شده شما می توانید به صفحه اختصاصی خود در سایت پیامک سامانه پیامک تمشک با آدرس فوق وارد شوید

گام دوم – شروع به کار با سرویس SOAP

سرویس SOAP IP از یکسری متد تشکیل شده است که هر یک برای منظوری خاص (ارسال پیامک ، دریافت وضعیت پیامک، دریافت میزان اعتبار باقیمانده و ..) طراحی شده اند . پس برای استفاده از SOAP کافیست که یکی از متدهای زیر را انتخاب نموده و پارامترهای مخصوص به آن متد را تنظیم نموده، سپس آن را از طریق نرم افزار خود به آدرس زیر فراخوانی کنید:

<http://sms.tameshksms.com/smsSendWebService.asmx?wsdl>

در ادامه شما می توانید تمامی متدهای SOAP و لیست پارامترهای مربوط به هر یک از متدها را مشاهده نمایید . برای آشنایی بیشتر، لیست کامل پارامترها به همراه توضیح هر کدام در انتهای توضیح متد ها قرار گرفته است :

◆ متد SendSms برای ارسال پیامک

◆ متد GetCredit برای دریافت میزان اعتبار باقی مانده

◆ متد GetReceiveSMS برای دریافت کل پیام های دریافتی از کل شماره های اختصاصی

♦ متد **GetReceiveSMSWithNumber** برای دریافت پیامک های دریافتی طبق یک شماره ، خوانده نشده و تاریخ شروع و

پایان

♦ متد **GetDelivery** برای دریافت وضعیت پیامک های ارسالی

♦ متد **GetCount** برای دریافت تعداد کل پیام های دریافتی از یک شماره اختصاصی

♦ متد **GetCountAll** برای دریافت تعداد کل پیام های دریافتی از کل شماره های اختصاصی

♦ متد **SendFutureSms** برای ارسال پیامک در آینده

♦ متد **CheckSendFuture** برای دریافت وضعیت ارسال پیام در آینده

توجه : برای شناخت هر چه بیشتر این سرویس و روش کار آن شما می توانید ا "نمونه کد" (Sample code) های آماده سیستمهای

پیام دهی نیز استفاده نمایید.

۱-متدها و پارامترها

۱-۱- متد **SendSms** : برای ارسال پیامک باید از این متد استفاده نمود . پارامترهای مربوط به این متد در جدول ذیل آورده شده

اند. مقدار بازگشتی این متد رشته ای است که مقادیر آن نشانگر شرایط پیامک می باشد. بدین ترتیب که اگر پارامترها صحیح

باشند و پیامک به صورت موفقیت آمیز در سیستم پیام دهی قرار می گیرد به ازای هر پیامک به صورت موفقیت آمیز در سیستم

پیام دهی قرار گیرد به ازای هر پیامک یک شناسه پیام (**SmsId**) و در غیر اینصورت کد خطای مربوطه در خروجی قرار می

گیرد .

۵

دفتر مرکزی : ساری – بلوار امیر مازندرانی – روبروی خیابان مهدی آباد – مجتمع زمرد طبقه چهارم

Website : Tameshksms.com

Email : info@tameshksms.com

Tel : ۰۱۱۲۳۳۹۴۸۵۴

sms center : ۰۵۰۰۰۵۷۰۵۰

توجه: لیست انواع خطاها و توضیحات مربوط به هر کدام در انتهای این راهنما آورده شده است :

SendSms			نام متد
ورودی			
نام پارامترها	نوع پارامترها	اجباری / اختیاری	توضیح
SmsText	String	اجباری	متن پیامک
SenderNumber	Array of String	اجباری	شماره فرستنده (شماره اختصاصی یا همان شماره ۳۰۰۰XXXX)
ReceiveNumber	String Array of String	اجباری	شماره گیرنده (شماره موبایل مقصد)
SendType	SendType	اجباری	روش ارسال
SmsMode	SmsMode	اجباری	روش ذخیره سازی در گوشی موبایل
توضیح : پارامترهای Username و Password و Domain از روی Header ارسالی سرویس SOAP خوانده می شوند.			
خروجی			
نوع پارامتر	توضیح		
String of Long integer	رشته ای است شامل شناسه پیام و یا کد خط		

۱ در هنگام ست کردن پارامتر متن به نکات زیر توجه فرمایید :

a) به طور کلی طول استاندارد هر پیامک ۱۴۰ بایت است . در نتیجه طول استاندارد یک پیامک فارسی ۷۰ کاراکتر (هر کاراکتر برابر ۲ بایت است)، اندازه استاندارد پیامک باینری ۱۴۰ بایت و طول استاندارد یک پیامک لاتین ۱۶۰ کاراکتر (۱۶۰ کاراکتر ۷ برابر ۲ بیتی) می باشد. باید به این نکته توجه داشته باشید که حتی وجود یک کاراکتر فارسی در پیامک باعث تغییر رمز نگاری آن پیامک شده و در نتیجه پیامک به طور کلی فارسی محاسبه می شود و طول آن بر اساس طول پیامک فارسی (۷۰ کاراکتر) در نظر گرفته خواهد شد.

(b) جهت ایجاد سهولت برای مشترکین امکان ارسال پیامک های بزرگتر از طول استاندارد فراهم گشته است و مشترکین می توانند بدون نیاز به شکستن متن های طولانی، آن ها را به صورت یکپارچه ارسال نمایند. فقط در هنگام استفاده باید توجه داشته باشید که به ازای هر بخش از یک پیامک فارسی طولانی، ۳ کاراکتر و به ازای هر بخش از یک پیامک لاتین طولانی، ۷ کاراکتر از فضای استاندارد متن پیامک به کد UDH اختصاصی پیدا می کند. یعنی در واقع هر بخش از پیامک چند قسمتی فارسی حداکثر ۶۷ کاراکتر و هر بخش از پیامک چند قسمتی لاتین حداکثر ۱۵۳ کاراکتر را در خود جای می دهد.

۲. شماره گیرنده یا شماره موبایل که باید به طور مثال به صورت زیر در متد تعریف شوند:

۰۹۱۲XXXXXXXX

۳. شماره فرستنده در هنگام ارسال می تواند به صورت زیر در متد تعریف شود:

۳۰۰۰XXXX

توضیح: برای ارسال پیامک هایی که طول آنها از طول استاندارد بیشتر است، سیستم به صورت خودکار پیامک را به تعداد بخش لازم تقسیم کرده و ارسال می نماید. لازم به توضیح است که میزان اعتبار کسر شده ز کاربر بر اساس تعداد بخش های متن پیام می باشد.

۴. با تنظیم پارامتر SmsMode می توان نحوه دریافت و نمایش پیامک ارسالی بر روی گوشی گیرنده را به دلخواه تغییر داد، مقادیر قابل قبول برای این پارامتر سه مقدار Alert, SaveInPhone, SaveInSim می باشد.

Alert: پیامک به صورت مستقیم بر روی صفحه موبایل شخص گیرنده فرستاده می شود. در این حالت پیامک در موبایل یا سیم کارت شخص گیرنده به صورت اتوماتیک ذخیره نمی شود و با خروج از آن از روی گوشی حذف می شود (پیامک خبری).

SaveInPhone: پیامک در حافظه موبایل شخص گیرنده ذخیره می شود (پیامک عادی)

SaveInSim: پیامک بر روی حافظه سیم کارت شخص گیرنده ذخیره می شود.

چند نکته مهم:

♦ **طول استاندارد پیامک:** سایز استاندارد پیامک ۱۴۰ بایت می باشد. در صورتیکه در متن پیامک از کاراکترهای Adphanumeric استفاده شود طول یک پیامک حداکثر می تواند ۱۶۰ کاراکتر و در صورتی که از Unicode دوبایتی (کاراکترهای فارسی) استفاده شود طول یک پیامک حداکثر ۷۰ کاراکتر می تواند باشد.

انواع حالت‌های ممکن ارسال پیامک از طریق متد SendSms :

♦ **ارسال پیام به صورت نظیر به نظیر:** در این حالت تمام پارامترهای متد باید دارای آرایه ای هم سایز و متناظر با آرایه پارامتر شماره دریافت کنندگان باشد. در این صورت می توان پیامک های گوناگونی به گیرندگان ارسال نمود. در این صورت باید پارامتر SendType از نوع DynamicText به تابع انتقال داده شود.

♦ **ارسال پیامک به صورت یک به چند (بالک):** این حالت تمام پارامترهای متد به جز پارامتر شماره دریافت کنندگان باید دارای آرایه ای به طول یک باشند. در این صورت یک متن واحد برای چندین نفر (شماره گیرنده) فرستاده می شود. در این صورت باید پارامتر SendType از نوع StaticText به تابع انتقال داده شود.

متد **GetCredit** : این متد برای گرفتن میزان اعتبار باقیمانده توسط کاربر است. پارامترهای مربوطه در جدول زیر آورده شده

است و مقدار برگشتی آن یک رشته می باشد و میزان اعتبار باقیمانده را بر حسب ریال نشان می دهد.

نام متد		GetCredit	
ورودی			
نام پارامتر	نوع پارامتر	اجباری/اختیاری	توضیح
توضیح: پارامترهای Username و Password و Domain از روی header ارسالی سرویس SOAP خوانده			
خروجی			
نوع پارامتر	توضیح		
String	اعتبار باقی مانده به ریال		

۱-۳ - متد **GetDelivery**: این متد برای گرفتن وضعیت واقعی پیامکهای ارسال شده توسط کاربر مورد استفاده قرار می گیرد. مقدار بازگشتی این متد یک مقدار صحیح شامل یکی از اعداد ۱ و ۲ و ۸ و ۱۶ و در غیر اینصورت عدد خروجی نشاندهنده نوع خطای صورت گرفته می باشد.

نام متد		GetDelivery	
ورودی			
نام پارامترها	نوع پارامترها	اجباری / اختیاری	توضیح
SmsId	Array of Long	اجباری	شناسه پیامک ها
خروجی			
نوع پارامترها		توضیح	
Array of integer		وضعیت پیامک ها تا این لحظه و یا کد خطا	

مقادیر خروجی این متد به صورت زیر می باشد:

- ◆ ۱:رسیده به گوشی
- ◆ ۲:نرسیده به گوشی
- ◆ ۸:رسیده به مخابرات
- ◆ ۱۶:نرسیده به مخابرات
- ◆ ۰:وضعیتی دریافت نشده است (یا پیامک در صف ارسال قرار دارد).

توضیح: شناسه پیامک از روش زیر به دست می آید:

۱. پس از اجرای دستور **SendSms** و در صورتیکه درخواست ارسال پیامک به صورت صحیح ارسال شود شناسه پیامک در خروجی این متد ظاهر خواهد شد.

۴-۱ متد **GetReceiveSMSWithNumber** : با استفاده از این متد کاربر می تواند پیامهای ورودی خود را دریافت نماید . مقادیر ارسالی به این متد شامل نوع فراخوانی ، شماره اختصاصی گیرنده، تعداد پیام ها، تاریخ شروع و تاریخ پایان است. در صورت انتخاب نوع فراخوانی **All** از مقدار تعریف شده **ReceiveType** ، نیازی به پر کردن مقادیر اختیاری نیست و خروجی تابع شامل کل پیامک های دریافتی می باشد. با انتخاب نوع فراخوانی **Top** و با وارد کردن مقدار **Count** تعداد مشخصی از پیام ها از ابتدا به صورت صعودی دریافت میشود. در این نوع فراخوانی مقدار **Count** به طور پیش فرض یک در نظر گرفته میشود. نوع فراخوانی **Down** مشابه فراخوانی **Top** است با این تفاوت که تعداد مشخصی از پیام ها را از آخر دریافت میکند. در صورت انتخاب نوع فراخوانی **WithDate** می توان پیام های دریافتی را برحسب تاریخ شروع و پایان به دست آورد.

GetReceiveSMSWithNumber			نام متد
ورودی			
نام پارامترها	نوع پارامترها	اجباری/ اختیاری	توضیح
RecType	ReceiveType	اجباری	نوع فراخوانی
LineNumber	string	اجباری	شماره اختصاصی گیرنده
Count	Integer	اختیاری	تعداد پیامهای مورد درخواست
FromDate	String	اختیاری	تاریخ شروع

تاریخ پایان	اختیاری	string	ToDate
			توضیح :
			خروجی
		توضیح:	نوع پارامترها
وب سرویس هستند		ReceiveSMS, ReceiveType از انواع تعریف شده	List of ReceiveSms

توضیح:

نوع خروجی ReceiveSms شامل :

- ReceiveNumber : شماره گیرنده

- SmsText : متن پیامک

- Date : تاریخ ارسال

- FromNumber : شماره ارسال کننده

- Error : شماره خطا

۱-۵- متد **GetReceiveSMS** : روش کار این متد مشابه متد فوق (**GetReceiveSMSWithNumber**) است با

این تفاوت که در این متد ، کاربر(درصورت داشتن چندین شماره اختصاصی) می تواند پیام های همه خطوط را دریافت نماید.

GetReceiveSMS			نام متد
ورودی			
نام پارامترها	نوع پارامترها	اجباری/اختیاری	توضیح
RecType	ReceiveType	اجباری	نوع فراخوانی
Count	Integer	اختیاری	
FromDate	string	اختیاری	تاریخ شروع
ToDate	String	اجباری	تاریخ پایان
توضیح :			
خروجی			
نوع پارامترها		توضیح	
List of ReceiveSms		receiveSMS و ReceiveType از انواع تعریف شده وب سرویس هستند	

۱-۶- متد **GetCountAll** : با فراخوانی این تابع کاربر می تواند تعداد کل پیامک های دریافتی از تمام شماره های اختصاصی خود را دریافت نماید.

نام متد		GetCountAll	
ورودی			
نام پارامترها	نوع پارامترها	اجباری/اختیاری	توضیح
توضیح :			
خروجی			
نوع پارامترها	توضیح		
Integer	خروجی این تابع تعداد پیامک های دریافتی و در صورت بروز خطا نشان دهنده شماره خطا است		

۷-۱- متد **GetCount** : با فراخوانی این تابع کاربر با وارد نمودن یک شماره اختصاصی می تواند تعداد کل پیامک های دریافتی از آن شماره را دریافت نماید.

GetCount			نام متد
ورودی			
نام پارامترها	نوع پارامترها	اجباری/اختیاری	توضیح
LineNumber	string	اجباری	
توضیح :			
خروجی			
نوع پارامترها	توضیح		
Integer	خروجی این تابع تعداد پیامک های دریافتی و در صورت بروز خطا نشان دهنده شماره خطا است		

۱-۸- متد **SendFutureSms** : با استفاده از این تابع کاربر می تواند در تاریخ و زمان خاصی اقدام به ارسال پیامک نماید. خروجی این تابع یک عدد است که در واقع شناسه این پیامک است که با ارسال آن به تابع **CheckSendFuture** می توان از وضعیت ارسال پیامک اطلاع حاصل کرد. مقادیر منفی خروجی تابع، نشان دهنده شماره خطای رخ داده است.

SendFutureSms			نام متد
ورودی			
توضیح	اجباری/اختیاری	نوع پارامترها	نام پارامترها
	اجباری	String	LineNumber
	اجباری	String	ReceiveNumber
	اجباری	String	Date
	اجباری	String	Time
	اجباری	string	SmsText
توضیح :			
خروجی			
	توضیح		نوع پارامترها
			integer

۹-۱- متد **CheckSendFuture** : با ارسال شناسه پیامک از وضعیت ارسال آن مطلع می شویم. شناسه پیامک، خروجی تابع **SendFutureSms** است. خروجی این تابع یک رشته است که نشان دهنده وضعیت پیامک ارسالی است.

نام متد		CheckSendFuture	
ورودی			
نام پارامترها	نوع پارامترها	اجباری/اختیاری	توضیح
FutureSmsId	integer	اجباری	
توضیح :			
خروجی			
نوع پارامترها	توضیح		
string			

توضیح خطاها:

نام کاربری یا رمز عبور اشتباه است	۱-
خطا در دریافت داده ها	۲-
شماره اختصاصی اشتباه است	۳-
فرمت داده های ورودی اشتباه است	۴-
شناسه پیامک اشتباه است	۵-
آرایه ورودی شناسه ها خالی است	۶-
فرمت تاریخ اشتباه است	۷-
تاریخ یا ساعت ورودی مربوط به گذشته است	۸-
وب سرویس برای این کاربر فعال نمی باشد	۹-
تعداد آرایه ورودی موبایل اشتباه است	۱۰-