## Check Credit

* Calls CheckCredit

  [WebMethod]

public CreditCheckResponse CheckCredit(

            Name billingName,

            string serviceZipCode,

            Contact contactInfo,

            PersonalCredentials billingContactCredentials,

            int numberOfLines,

            string question,

            string answer,

            string referenceNumber)

        {

            var request = new CheckCreditRequest

                              {

                                  BillingName = billingName,

                                  ServiceZipCode = serviceZipCode,

                                  ContactInfo = contactInfo,

                                  BillingContactCredentials = billingContactCredentials,

                                  NumberOfLines = numberOfLines,

                                  QuestionCode = question,

                                  QuestionAnswer = answer,

                                  ReferenceNumber = referenceNumber

                              };

            var impl = new CheckCredit();

            return impl.Execute(request, referenceNumber);

        }

## CheckCreditExistingAccount

* Calls CheckCreditExistingAccount

   [WebMethod]

        public object CheckCreditExistingAccount(

            string accountNumber,

            string secretPin,

            string question,

            string answer,

            string ssn,

            CustomerType customerType,

            OrderType orderType,

            byte handsetCount,

            string referenceNumber)

        {

            var req = new CheckCreditExistingAccountRequest

                          {

                              AccountNumber = accountNumber,

                              SecretPin = secretPin,

                              CustomerType = customerType,

                              HandsetCount = handsetCount,

                              ReferenceNumber = referenceNumber,

                              OrderType = orderType,

                              QuestionAnswer = answer,

                              QuestionCode = question,

                              SSN = ssn

                          };

            var impl = new CheckCreditExistingAccount();

## CustomerLookupBy MDN

* Calls:
  + CustomerLookupByMdn
  + CheckCreditExistingAccount

    [WebMethod]

        public CustomerInquiryResponse CustomerLookupByMDN(string mdn, string secretKey, string ssn, string questionAnswer, string referenceNumber)

        {

            var resp = new CustomerInquiryResponse();

            var req = new CustomerInquiryRequest

                          {

                              Mdn = mdn,

                              SecretKey = secretKey,

                              ReferenceNo = referenceNumber,

                              QuestionAnswer = questionAnswer,

                              Ssn = ssn

                          };

            var impl = new CustomerLookupByMdn();

            resp = impl.Execute(req, referenceNumber);

            if (resp != null && resp.ErrorCode == 0)

            {

                if (resp.ServiceResponseSubCode != (int)ServiceResponseSubCode.CL\_CUSTOMER\_NOT\_FOUND

                    && resp.ServiceResponseSubCode != (int)ServiceResponseSubCode.CL\_ACCOUNT\_NOPIN

                    && resp.ServiceResponseSubCode != (int)ServiceResponseSubCode.CL\_ACCOUNT\_LOCKED

                    && resp.ServiceResponseSubCode != (int)ServiceResponseSubCode.CL\_INVALID\_PIN)

                {

                    // total lines approved & address data need to be retrieve from credit check web service call

                    var ccReq = new CheckCreditExistingAccountRequest

                                    {

                                        AccountNumber = resp.CustomerAccountNumber,

                                        SecretPin = secretKey,

                                        SSN = ssn,

                                        QuestionAnswer = questionAnswer,

                                        // **TODO need confirmation from Sprint on this.**

                                        CustomerType = CustomerType.INDIVIDUAL,

                                        HandsetCount = 1,

                                        ReferenceNumber = referenceNumber,

                                        OrderType = OrderType.ADD\_ON

                                    };

                    var ccImpl = new CheckCreditExistingAccount();

                    var ccResp = ccImpl.Execute(ccReq, referenceNumber);

### NpaLookupByZip

[WebMethod]

        public NpaResponse NpaLookupByZip(string zipCode, string referenceNumber)

        {

            return ServiceImplementation.NpaLookupByZip.Execute(zipCode, referenceNumber);

        }

## NpaNxx

[WebMethod]

        public NpaResponse NpaNxx(string zipCode, string referenceNumber)

        {

            var req = new NpaRequest { Zip = zipCode, ReferenceNumber = referenceNumber };

            var impl = new NpaNxx();

            return impl.Execute(req, referenceNumber);

        }

## ResendPin

  [WebMethod]

        public WAAuthenticationResponse ResendPin(string Mdn, string referenceNumber)

        {

            var req = new WAAuthenticationRequest { MDN = Mdn, ReferenceNo = referenceNumber };

            var impl = new AuthenticationResend();

            return impl.Execute(req, referenceNumber);

        }

## SubmitOrder

 [WebMethod]

        public OrderResponse SubmitOrder(string OrderNumber)

        {

            var req = new WaActivationRequest(short.Parse(OrderNumber));

            var impl = new Activation();

            return impl.Execute(req, req.CheckoutReferenceNumber);

        }

## ValidatePortIn

  [WebMethod]

        public ValidatePortInResponse ValidatePortIn(List<MDNSet> mdnList, string referenceNumber)

        {

            var req = new PortEligibilityRequest { MDNList = mdnList, ReferenceNumber = referenceNumber };

            var impl = new PortEligibility();

            return impl.Execute(req, referenceNumber);

        }