## Изисквания към интеграцията, сигурността и качеството на системи в рамките на електронното управление

# Съдържание

1. [Цел на документа](#1-Цел-на-документа)
2. [Текущо състояние](#2-Текущо-състояние)
3. [Предложени решения](#3-Предложени-решения)
4. [Изисквания](#4-Изисквания)
5. [Общи бележки](#5-Общи-бележки)
6. [Често задавани въпроси](#6-Често-задавани-въпроси)

### 1. **Цел на документа**

Целта на този документ е да представи конкретни технически параметри на системите, с цел реализиране на електронно управление. Изискванията ще се прилагат при разработване на нови и при надграждане на съществуващите системи.

Документът, техническите детайли към него и историята на редакциите му могат да бъдат намерени и на адрес <http://dev.egov.bg> (все още не функционира). Коментари и заявки за промени на документа могат да бъдат изпращани на <https://github.com/governmentbg/egov-requrements/issues>.

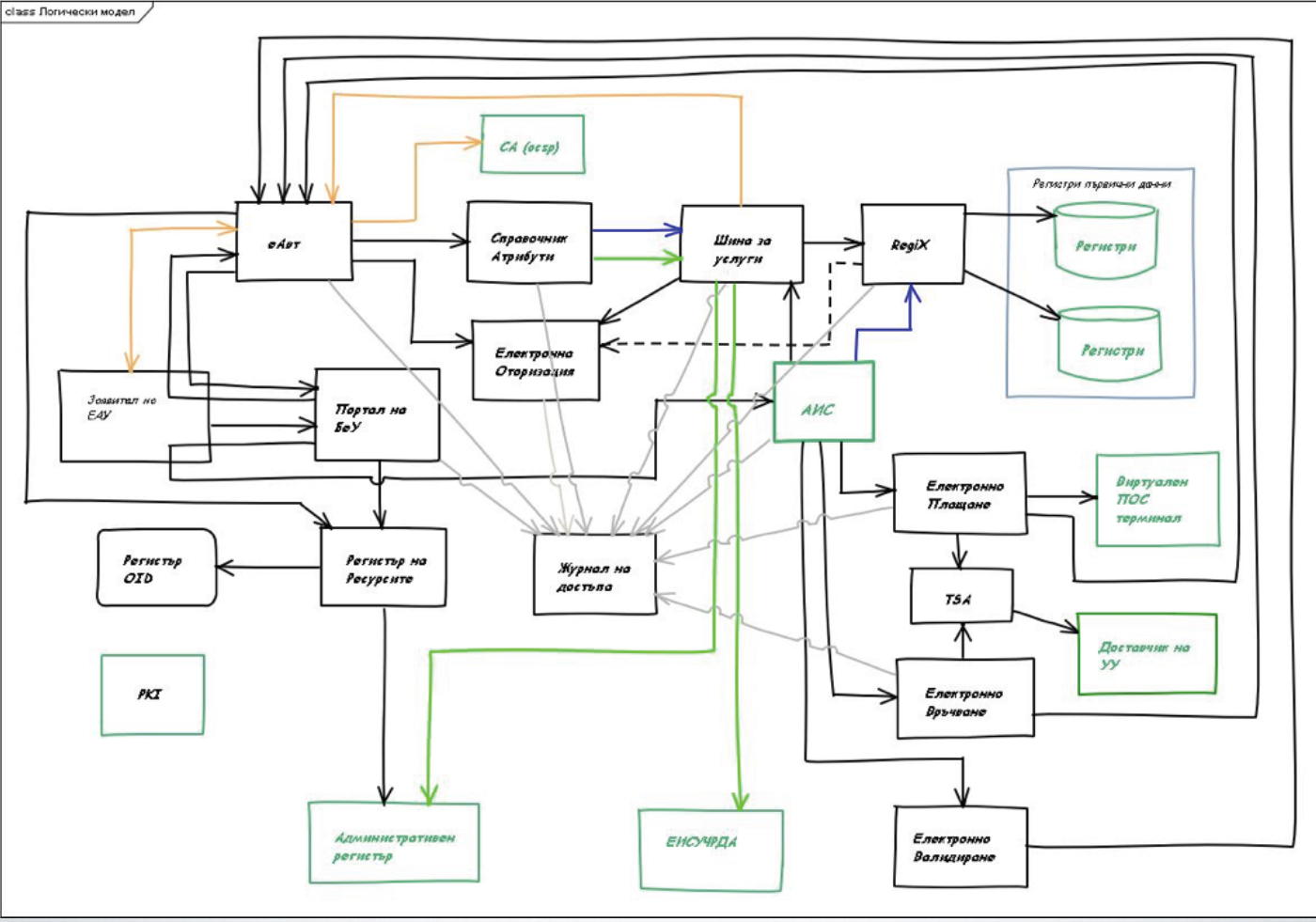
### 2. **Текущо състояние**

В момента електронното управление де-факто не функционира заради липса на интеграция между системите. Това се дължи на три фактора:

* нормативен - нормативната уредба (Законът за електронно управление и наредбите към него) не позволяват свободна интеграция между системи. Вместо това изискват използване на де-факто “умно” ESB (ЕСОЕД - Единна среда за обмена на електронни документи), през което да минава цялата комуникация. Освен това достъпът до данните на първичните администратори на данни, макар отворен според ЗЕУ (Закон за електронното управление), е изрично затворен според специални закони.
* организационен - за да бъде пусната една система в експлоатация, цялата нейна изходяща и входяща комуникация трябва да бъде одобрена от Съвет по вписванията, и да бъде вписана в Регистрите за оперативна съвместимост. Освен това получаването на достъп до даден регистър преминава през специални споразумения и допълнителни стъпки. Липсва и желание от страна на някои администрации да бъдат интегрирани.
* технологичен - текущата комуникация между системи и регистри е на база на “електронна услуга” и “електронен документ”. Това не предоставя необходимата грануларност на достъп до данните, което създава правни пречки. Освен това ЕСОЕД и RegiX (система за свързване на регистри чрез уеб-услуги) не притежават някои необходими функционалности (напр. RegiX не поддържа асинхронни и subscribe заявки). Комуникацията може да минава и през “шина за услуги” (ESB). Първичните регистри, от своя страна, са неподготвени да посрещнат натоварването, което следва да понесат при реално функциониране на електронното управление.

Макар на теория настоящият набор от нормативни, организационни и технологични решения да изглежда добър, той се оказва неприложим на практика.

Следва графика на настоящата архитектура. За съжаление към момента тя не функционира.

 (източник: презентация на BUL-SI по проект на МТИТС)

От всичко изложено дотук не следва премахването на съществуващата система, но е необходимо опростяване на някои нейни компоненти и взаимодействия. Много от описаните по-долу компоненти са вече реализирани до определено ниво, и могат да се преизползват или надградят. Например „журнал на достъпа“, „регистър на услугите“, RegiX и др.

Описаните по-долу интерфейси са предвидени не непременно да заменят съществуващи решения, а да позволят лесната комуникация между системи, при която изпълнителите на системите да разчитат единствено на публична документация за да бъдат интегрирани, а не на действия и формални решения от административни органи.

### 3. **Предложени решения**

Нормативните проблеми са вече адресирани със закона за изменение и допълнение на ЗЕУ. Елиминира се нуждата от централизиран подход чрез ЕСОЕД (макар такъв да остава възможен), елиминират се нуждите от споразумения, където това е било нормативно необходимо към момента. Регламент 910/2014 пък разширява тълкуването на понятието “електронен документ”, така че да включва всички данни в електронен вид.

Организационните и технологичните проблеми ще бъдат адресирани в този документ, като на повечето организационни проблеми ще бъдат предоставени технологични решения, елиминиращи човешкия фактор.

Предложеното се базира на Архитектурата за електронно управление в [Приложение 1](https://docs.google.com/document/d/1WEbNObrBxu1SmpBcq27OzIiCuDM8vdn-xvlBk55SBJA/edit?usp=sharing).

Изискванията ще се прилагат към всички проекти за изграждане и надграждане на първични регистри и административни информационни системи, финансирани както по ОПДУ (Оперативна програма “Добро управление”), така и от републиканския бюджет. По този начин ще бъде осигурено плавното преминаване към реално функционираща система на електронното управление, в която всички системи, които имат нужда да комуникират помежду си, ще могат да го правят без организационни, технологични и нормативни пречки.

Предложената архитектура и технологични решения следват принципите KISS и YAGNI (Keep it simple, stupid и You ain’t gonna need it). Т.е. целта е максимално да се опрости интеграцията и изграждането на системи.

В допълнение, по изискванията на оперативната програма, всички системи ще бъдат разработвани с отворен код, като по този начин преизползването на често срещани функционалности ще бъде улеснено.

### 4. **Изисквания**

Изисквания към интеграцията  
Изисквания към качеството

### 5. **Общи бележки**

* Освен интерфейсите, описани в изискванията за интеграция, всеки АИС (вкл. регистър) може да поддържа и произволни уеб-услуги. Проверката на автентикацията на заявителя е отговорност на системата, предоставяща уеб-услугата.
* Всички структури на данни за комуникация между АИС, първични регистри и координатор могат да бъдат намерени на [*http://dev.egov.bg/schemas*](http://dev.egov.bg/schemas), както и на адрес на всеки АИС (вж секция “Документация на интерфейсите)
* Форматът за дати и часове следва да бъде ISO 8601
* Частните ключове, използвани за електронно подписване и декирптиране, следва да бъдат съхранявани на HSM
* Всяка операция трябва да бъде повторена при неуспех минимум 3 пъти, с т.нар. exponential backoff.
* Ако и след повторението изпращането на отговор към callbackUrl не е успешно, трябва да се запише като “изчакващо” в системата и да бъде изпратено в следващ момент.
* Рутерите, firewall-ите и други механизми следва да бъдат конфигурирани така, че да позволяват използването на нестандартни HTTP header-и (като Signature)
* Според условията за допустимост на ОПДУ и аналогично РМС, всички системи следва да се разработват с отворен код от ден едно в публично хранилище.

### 6. **Често задавани въпроси**

* Защо не се използва изцяло RegiX?
* RegiX адаптира стари (legacy) системи. В тази си част той ще бъде използван. В случаите, в които първичните регистри, които RegiX адаптира биват надградени, става излишно и усложняващо поддържането на адаптер в синхрон с промените в системата и структурата на нейните данни. Поддържането на SUBSCRIBE функционалност чрез адаптер е трудно постижима.
* Защо не се използва oid регистър, а всеки АИС публикува избран от себе си идентификатор?
* Ръчното поддържане на още един регистър, с процедурите по вписване в него, утежнява процедурата по включване. Предвид, че всеки АИС все пак принадлежи на някоя администрация, която е вписана в ИИСДА, идентификаторите в рамките на администрацията следва да се избират от нея и да не бъдат одобрявани от никого.\*
* Защо peer-to-peer комуникация?
* Поддържането на сложен централен интеграционен компонент има смисъл единствено, когато стари (legacy) системи трябва да бъдат интегрирани. Предвид престоящото надгражене и изграждане на първични регистри, такъв интеграционен компонент не е нужен. Още повече, че той би следвало да поема цялото натоварване, което го прави по-сложен за реализиране и поддържане. При peer-to-peer комуникацията всяка система, която има нужда от дадени данни (и има законово основание да ги чете), ще може да направи това с проста уеб-услуга.
* Защо все пак е нужен централен координатор на заявки?
* Централен координатор (който обаче не е ESB и няма за цел да трансформира или рутира заявки) е необходим заради проверките на правото на достъп и за запзване на централен журнал на достъпа (на база на който да се установява нерегламентиран достъп, например). Също така централният компонент поема и автентикацията на заявителя пред регистъра. Така нито заявителят, нито регистърът има нужда да реализират функционалност за проверка на права на достъп, за запис в журнал или за автентикация. Единственото, което следва да направят е по една заявка към централния координатор.
* Защо се използва PKI за автентикация на системи?
* Теоретично е възможно използването на HMAC за автентикация при правене на заявка. Както е описано [тук](http://crypto.stackexchange.com/questions/5646/what-are-the-differences-between-a-digital-signature-a-mac-and-a-hash)), HMAC автентикацията не предоставя уверение за пред 3-ти лица, че заявката наистина е изпратена от този, който твърди, че я изпраща. Т.е. компрометиран централен компонент би могъл да инициира заявка от името на кой да е заявител. Използването на електронен подпис и PKI (в технологичния смисъл), решава този проблем.
* Защо отпада съгласуването на заявките, отговорите и структурите от данни за оперативна съвместимост
* Съгласуването на схеми на документи е излишна бюрократична стъпка. Първичните администратори на данни, като отговорници за данните, дефинират формата, който се използва. Всички останали консуматори следват така дефинирания формат. Все пак, вече дефинираните и одобрени структури следва да се използват при надграждането на регистрите, а органът, отговорен за електронното управление, следва да консултира форматите с изпълнителите, изграждащи информационните системи на регистрите.
* Защо човешкият фактор се минимизира?
* Всяко човешко действия в процеса въвежда потенциал за грешка, за забавяне или дори за корупция. Затова свеждането им до минимум и използването на системата на “автопилот” е ключово за бързото и качествено реализиране на електронното управление. Човешкият фактор се свежда до:
  + въвеждане на данните, до които услугите имат достъп и тяхното правно основание
  + одобряване на регистрираните администрации, след което да им бъде издаден цифров сертификат.