



ЕВРОПЕЙСКИ СЪЮЗ  
ЕВРОПЕЙСКИ ФОНД ЗА  
РЕГИОНАЛНО РАЗВИТИЕ



ОПЕРАТИВНА ПРОГРАМА  
ИНОВАЦИИ И  
КОНКУРЕНТОСПОСОБНОСТ

## ТЕХНИЧЕСКА СПЕЦИФИКАЦИЯ

По процедура за определяне на изпълнител чрез избор с публична покана с предмет:

**„Закупуване на дълготрайни нематериални активи:**

- Софтуерен компонент PPSP медиен енкодер (Инжектор) - 1 бр;

- Софтуерен компонент PPSP супер клиент (Super Peer) - 1 бр;

- HTML5 Видео Плеър (HTML 5 Player) - 1 бр;

- Софтуер за управление на мултимедийната платформа (Облачен Мениджмънт панел) - 1 бр.”,

по проект „Подобряване на производствения капацитет в „Скайникс“ ЕООД, по договор за БФП № BG16RFOP002-2.040-1479-C01, финансиран от Оперативна програма „Иновации и конкурентоспособност“, съфинансирана от Европейския съюз чрез Европейския фонд за регионално развитие, както следва:

### **А) Минимални технически и функционални характеристики:**

*/Минималните технически и функционални характеристики са задължителни за offerиране от кандидатите. Оценителната комисия ще отстранява от по-нататъшно участие в търга кандидати, в чиито offerти за оборудване не фигурират посочените минималните технически и функционални характеристики./*

№	Наименование на актива	Количество	Минимални технически и функционални характеристики
1.	Софтуерен компонент PPSP медиен енкодер (Инжектор)	1 брой	<p>Софтуерно приложение написано на C/C++. Трябва да може да се изпълни и в Докер контейнер под Линукс.</p> <p><b>Основни характеристики:</b></p> <p>Имплементация на протокол PPSP, по RFC7574. Специфичен LEDBAT алгоритъм за предоставяне на congestion control. Използване на UDP съобщения за транспорт. Взаимодействие с тракер или някакъв друг механизъм за обмяна на пиъри. Да поддържа различни транспортни протоколи за източник на данни- RTSP, HTTP/HTTPS, Multicast streaming, Unicast streaming, HLS, MPEG-DASH.</p> <p><b>Multiplexor-</b> да има възможност да комбинира няколко стрийма</p> <p><b>Поддръжка на известните видео кодеци-</b> H.264, H.265, VP9/10</p> <p><b>Поддръжка на известните аудио кодеци-</b> AAC, MP3</p> <p><b>Видео контейнери-</b> MPEG, MPEG-2, MPEG2-TS</p> <p><b>Demultiplexor-</b> да може да филтрира някои стриймове и програми</p> <p><b>Връзка с native WebRTC-</b> трябва да използва най-новите библиотеки от native WebRTC за връзка с различните уеб браузъри.</p> <p><b>Използване на webSockets</b> – За връзка със Skynix Cloud System (SCS)</p>

----- [www.eufunds.bg](http://www.eufunds.bg) -----

Този документ е създаден с финансовата подкрепа на Оперативна програма „Иновации и конкурентоспособност“ 2014-2020, съфинансирана от Европейския съюз чрез Европейския фонд за регионално развитие. Цялата отговорност за съдържанието на документа се носи от „Скайникс“ ЕООД и при никакви обстоятелства не може да се приема, че този документ отразява официалното становище на Европейския съюз и Управляващия орган.



2.	Софтуерен компонент PPSP супер клиент (Super Peer)	1 брой	<p>Софтуерно приложение написано на C/C++. Трябва да може да се изпълни и в Docker контейнер под Линукс.</p> <p><b>1. Основни характеристики</b> Имплементация на протокол PPSP, по RFC7574. Поддръжка на стрийминг и VOD услуги. Специфичен LEDBAT алгоритъм за предоставяне на congestion control. Използване на UDP съобщения за транспорт. Взаимодействие с тракер или някакъв друг механизъм за обмяна на пиъри. Осигурява правилното функциониране на суормовете</p> <p><b>2. PPSP component-</b> да осъществява връзка между участниците и инжектора. Изграждане на пиър клиент върху WebRTC, спрямо спецификацията RFC 7574 PPSP, използвайки RTCPeerConnection и RTCDDataChannel. Да може да приема и създава RTCDDataChannel връзка към други клиенти в мрежи с реални IP адреси и използващи NAT и по специално Full-cone, Address-restricted-cone, Port-restricted-cone NAT. Част от WebRTC спецификацията гарантира поддръжката и нормалната работа в такива мрежи чрез използване на STUN/TURN/ICE услуги. Осигуряване на сигурна, криптирана връзка между участниците</p> <p><b>3. Използване на webSockets</b> – За връзка със Облачния мениджмънт панел.</p>
3.	HTML5 Видео Плеър (HTML 5 Player)	1 брой	<p><b>HTML5 Player</b> трябва да бъде HTML5 Мултимедиян плеър, съвместим с най-популярните Интернет браузъри и да бъде съставен от следните компоненти;</p> <p><b>1. HTML5 видео плеър (HTML5 Player)</b> Да поддържа аудио и видео до поискване както и поточно аудио и видео. Пълна поддръжка на HTML5 видео стандарта с поддръжка на съответните видео и аудио формати и кодеци. Да ползва Media Source Extensions (MSE) за възпроизвеждане на съдържанието.</p> <p><b>2. Демултиплексор/Мултиплексор (Transmux) компонент</b> – да инспектира и при нужда да променя входните аудио/видео контейнер формати с цел да ги направи съвместими за възпроизведена на поддържаните браузъри чрез вградения HTML5 видео таг. Да поддържа следните входни контейнери и кодеци – MPEG-2 Transport Stream и AAC/MP3. Възможност за разделяне на видео и аудио потоци за нуждите на <b>HTML5 Player</b> ползвайки различни SourceBuffer. Изходния формат да бъде ISO BMFF(fMP4). Да използва Web Workers за демултиплексиране и мултиплексиране на аудио и видео потоците. Комуникацията между главната и подчинената нишка да стават чрез TransferableObjects.</p> <p><b>3. PPSP компонент</b> – Peer-To-Peer компонент за осъществяване на p2p комуникация и предаване на данни с други клиентски браузъри ползвайки WebRTC, компонент за осъществяване на връзка с облачна услуга signalling-tracker ползвайки Web Sockets API на браузъра. Изграждане на пиър клиент върху WebRTC спрямо спецификацията RFC 7574 PPSP, използвайки RTCPeerConnection и RTCDDataChannel. Изграждане на тракер клиент който поддържа signalling механизъм за осъществяване на връзки към други клиенти чрез WebRTC и tracker ползвайки WebSocket протокол. Да може да приема и създава RTCDDataChannel връзка към други клиенти в мрежи с реални IP адреси и използващи NAT и по</p>

----- [www.eufunds.bg](http://www.eufunds.bg) -----



			специално Full-cone, Address-restricted-cone, Port-restricted-cone NAT. Осигуряване на сигурна, криптирана връзка между участниците.
4.	Софтуер за управление на мултимедийната платформа (Облачен Мениджмънт панел)	1 бр.	<p>Многопотребителски облачен панел за управление. Трябва да бъде изпълнен ползвайки архитектурен шаблон – микроуслуги. Всяка услуга да може стартира като докер контейнер в Кубернетис клъстер. Трябва да бъдат изработени следните микроуслуги:</p> <p><b>- UserService</b> Да поддържа 2 типа акаунти – потребители и администратори. Ауторизация и автентикация чрез OAuth2. Скалируем клъстер база данни. Трябва да поддържа RestAPI програмен интерфейс. Трябва да поддържа асинхронна комуникация с другите микроуслуги чрез KafkaService. US трябва да съхранява чувствителните потребителски данни в криптиран вид.</p> <p><b>- SwarmService:</b> - да предоставя възможност за съхранение на информация и различни параметри свързани с Live Stream съдържанието. Съхраняване на конфигурацията и нейното предаване при заявка от Injector или SuperPeer. Да съхранява подадената информация в скалируема база данни</p> <p><b>- SignallingService</b> Да комуникира с участниците чрез връзка с WebSocket Протокол Да мвнажерира комуникацията между клиенти според WebRTC спецификацията. Да комуникира с останалите чрез Apache Kafka. Да проверява автентикацията на всеки външен участник.</p> <p><b>- TrackerService</b> Да обменя информация между себе си и външните клиенти през Apache Kafka към и от Signalling Service-a. Да отговаря на спецификацията RFC 7846 PPSTP. Да събира информация за участниците в PPSP мрежата при поискване от друг микросървис.</p> <p><b>- BillingService</b> Клиентът да има възможност да <b>избира и променя билинг план</b>. Интеграция на BS с <b>online payment processing platform</b>, с възможност за автоматични периодични плащания от клиента. <b>Изготвяне на фактура</b> след края на всеки отчетен период в PDF формат и изпращане по електронна поща до клиента. <b>Осигуряване на справочна информация:</b> трафик за отчетния период, информация за потреблението по суормове, справка за издадени фактури; Достъпът на <b>администратора и на клиентите до BS</b> да се осъществява чрез REST API с криптирана връзка и задължителна автентикация.</p> <p><b>- StatsService</b> Възможност да <b>предоставя в реално време данни</b> за статуса</p>

----- [www.eufunds.bg](http://www.eufunds.bg) -----



			<p>на инжекторите и суормовете, броя пиъри и битрейта на всеки суорм, системна информация за инжекторите. Данните, които клиента получава, да са в <b>JSON формат</b>.</p> <p><b>- ClientUI</b> Централизиран Angular уеб панел за достъп и управление на системата от крайни клиенти посредством потребителско име и парола. Панел за въвеждане и редактиране на данните на потребителя. Панел за настройки на използваните домейни, инжектори, суормове, API KEY. Панел за заплащане на услугата. Панел за информация за потреблението – клиенти, трафик, приходи, състояние на системата, статус на суормовете и др.</p> <p><b>- AdminUI</b> Angular Уеб панел за достъп и управление на системата от администратор. <b>Възможност</b> за Менажиране на потребителските акаунти. Създаване и менажиране на начините за заплащане от потребителите. Създаване и управление на планове за потребление. Показване статистики за отделните потребители, потребление на ресурси. Да показва състояние на системата и микроуслугите. Система за нотификация при закъснели плащания от потребители.</p> <p><b>- EmailService</b> Да позволява абониране за съобщения от следните микроуслуги – UserService, BillingService, SwarmService. Да изпраща писма до потребителите на платформата чрез SendGrid WebAPI. Трябва да поддържа RestAPI програмен интерфейс за изпращане на имейли. Трябва да поддържа асинхронна комуникация с другите микроуслуги чрез KafkaService.</p>
--	--	--	--

#### Б) Допълнителни технически изисквания:

/Допълнителните технически изисквания не са задължителни за офертиране от кандидатите. Оценителната комисия няма да отстранява кандидати, в чиито оферти за оборудване не фигурират посочените допълнителните технически изисквания/

№	Наименование на актива	Количество	Минимални технически и функционални характеристики
1.	Софтуерен компонент PPSP медиен енкодер (Инжектор)	1 брой	<p><b>1.1. PushStreaming</b> – Да бъде вграден RTMP сървър, който да приема Push стрийминг.</p> <p><b>1.2. SocialPublishing</b> – След получаване на команда от SocialService със source и credentials инициира публикуване на лайв стрийм към Youtube и Фейсбук.</p> <p><b>1.3. Изпращане на подробни статистики към Skynix Cloud System (SCS)</b> - битрейт, свързани пиъри, общо количество свалени и качени данни, време на живот на стриймъра, използвана памет, свободна памет, използвано място, свободно място, използван процесор freeRam, freeStorage, cpu, procVirtMem, ramUsedByProcess, totalRam, totalStorage.</p> <p><b>1.4. Получаване на конфигурацията дистанционно от Skynix Cloud System (SCS) през WebSocket</b> - конфигурацията</p>

----- [www.eufunds.bg](http://www.eufunds.bg) -----

Този документ е създаден с финансовата подкрепа на Оперативна програма „Иновации и конкурентоспособност“ 2014-2020, съфинансирана от Европейския съюз чрез Европейския фонд за регионално развитие. Цялата отговорност за съдържанието на документа се носи от „Скайникс“ ЕООД и при никакви обстоятелства не може да се приема, че този документ отразява официалното становище на Европейския съюз и Управляващия орган.



			трябва да съдържа подробна информация за swarm (peerCount, maxPeerCount, liveDiscardWindow, bufferType, nChunksPerSig, chunkSize, latestBin, source, ContentType, merkleHashFunction, contentProtectionMethod, liveSignatureAlgorithm, chunkAddrMethod, sourceUrl, sourceType) nChunksPerSig, chunkSize, latestBin, sourceContentType, merkleHashFunction, contentProtectionMethod, liveSignatureAlgorithm, chunkAddrMethod, sourceUrl, sourceType) chunkAddrMethod, sourceType.
2.	Софтуерен компонент PPSP супер клиент (Super Peer)	1 брой	<p><b>2.1. Изпращане на статистики към Skynix Cloud System (SCS)</b> - битрейт, свързани пиъри, общо количество свалени и качени данни, време на живот на стриймъра, използвана памет, свободна памет, използвано място, свободно място, използван процесор.</p> <p><b>2.2. Получаване на конфигурацията дистанционно от Skynix Cloud System (SCS) през WebSocket</b> - конфигурацията трябва да съдържа подробна информация за swarm (peerCount, maxPeerCount, liveDiscardWindow, bufferType, nChunksPerSig, chunkSize, latestBin, sourceContentType, merkleHashFunction, contentProtectionMethod, liveSignatureAlgorithm, chunkAddrMethod, sourceUrl, sourceType)</p>
3.	HTML5 Видео Плеър (HTML 5 Player)	1 брой	<b>НЕПРИЛОЖИМО!</b>
4.	Софтуер за управление на мултимедийната платформа (Облачен Мениджмънт панел)	1 брой	<p><b>4.1. StunService</b> Автономна микрослуга представляваща STUN сървър, изпълнен по RFC 3489. Осигурява "Network address translation traversal" и позволява успешна комуникация на пиъри в частни мрежи зад „NAT“.</p> <p><b>4.2. LogoService</b> Микрослуга за съхранение на картинки, използва се за менажиране на лого на даден канал чрез ClinetUI. Предоставя REST интерфейс, чрез който става менажирането на логата.</p> <p><b>4.3. SocialService</b> Микрослуга управляваща публикуването на съдържанието на живо към социални мрежи Фейсбук и Ютуб.</p> <p><b>4.4. StreamerService</b> - Микрослуга, която управлява създаването, спирането, стартирането и изтриването на облачно базирани стриймъри. Командите за извършване на посочените действия се получават като Kafka съобщения от: - Swarm service – за създаване и премахване на стриймъри; - Admin service – за спиране и стартиране на стриймъри (например за спиране на всички облачни стриймъри на потребител с деактивиран акаунт и др.) Получените съобщения се валидират и изпращат чрез websocket връзка към съответния Streamer Worker, който генерира съответната команда и я изпълнява в Kubernetes.</p>

----- [www.eufunds.bg](http://www.eufunds.bg) -----

Този документ е създаден с финансовата подкрепа на Оперативна програма „Иновации и конкурентоспособност“ 2014-2020, съфинансирана от Европейския съюз чрез Европейския фонд за регионално развитие. Цялата отговорност за съдържанието на документа се носи от „Скайникс“ ЕООД и при никакви обстоятелства не може да се приема, че този документ отразява официалното становище на Европейския съюз и Управляващия орган.



ЕВРОПЕЙСКИ СЪЮЗ  
ЕВРОПЕЙСКИ ФОНД ЗА  
РЕГИОНАЛНО РАЗВИТИЕ



ОПЕРАТИВНА ПРОГРАМА  
ИНОВАЦИИ И  
КОНКУРЕНТОСПОСОБНОСТ

			<b>4.5. StreamerWorker</b> - Микроуслуга, представляваща Kubernetes client, чрез която се изпращат командите от Streamer Manager към Kubernetes. Във всеки data center, в които се създават стриймъри, работи по една инстанция на Streamer Worker, която при стартиране се регистрира в Streamer Manager и започва да чака команди по websocket.
--	--	--	---

**Място за доставка: гр. Стара Загора, п.к. 6000, бул. „Славянски“ № 22, бизнес център „Практис“, ет. 3.**

----- [www.eufunds.bg](http://www.eufunds.bg) -----

Този документ е създаден с финансовата подкрепа на Оперативна програма „Иновации и конкурентоспособност“ 2014-2020, съфинансирана от Европейския съюз чрез Европейския фонд за регионално развитие. Цялата отговорност за съдържанието на документа се носи от „Скайникс“ ЕООД и при никакви обстоятелства не може да се приема, че този документ отразява официалното становище на Европейския съюз и Управляващия орган.