



I E G U L D Ī J U M S T A V Ā N Ā K O T N Ē

**Iegultās elektronikas programmēšanas kompetences paaugstināšanas apmācības
IEPIRKUMS**
(iepirkuma identifikācijas Nr. 6-8/A-29)

Pasūtītājs:

Nosaukums:	Biedrība „Latvijas Elektrotehnikas un elektronikas rūpniecības asociācija”
Reģistrācijas Nr.:	40008010789
Adrese:	Dzirnavu iela 93, Rīga, LV-1011, Latvija
Interneta adrese:	www.letera.lv
Kontaktpersona:	Dace Uškeneka
Tālrunis:	67288360
E-pasts:	apmacibas@letera.lv

Iepirkums tiek veikts saskaņā ar:

- 2017. gada 28. februāra Ministru kabineta noteikumiem Nr.104 „Noteikumi par iepirkuma procedūru un tās piemērošanas kārtību pasūtītāja finansētiem projektiem”,
- 2015. gada 27. oktobra Ministru kabineta noteikumiem Nr.617 “Darbības programmas "Izaugsme un nodarbinātība" 1.2.2. specifiskā atbalsta mērķa "Veicināt inovāciju ieviešanu komersantos" 1.2.2.1. pasākuma "Atbalsts nodarbināto apmācībām" pirmās projektu iesniegumu atlases kārtas īstenošanas noteikumi.”

Iepirkums tiek veikts Darbības programmas "Izaugsme un nodarbinātība" 1.2.2.specifiskā atbalsta mērķa “Veicināt inovāciju ieviešanu komersantos” 1.2.2.1.pasākuma “Atbalsts nodarbināto apmācībām” projekta Nr. 1.2.2.1/16/A/011 “Elektronikas, optikas, elektrotehnikas un telekomunikācijas uzņēmumu darbinieku prasmju pilnveidošana” ietvaros, sadarbībā ar Centrālo finanšu un līgumu aģentūru (CFLA).

IEPIRKUMA PRIEKŠMETA APRAKSTS

Mācību kurs Nr.1

Mācību kursa nosaukums:	C++ Programming for Embedded Systems
Mācību kursā obligāti ietveramas tēmas:	<p>Learn about the differences between C and C++.</p> <p>Requirements for C++ in an Embedded Systems:</p> <ul style="list-style-type: none">- Embedded system characteristics;- Language choice;- Memory mapped peripherals;- Volatile variables;- Compilation;- System boot-up;- Best Practices. <p>Summary of C:</p> <ul style="list-style-type: none">- A refresher on basic C syntax;- Functions;- Control flow;- Fundamental types;- Literals;- Derived types;- Operators;- Standard libraries. <p>From C to C++ and C++11:</p> <ul style="list-style-type: none">- The features added to C by C++ and the ANSI C-1999 standard;- Inline;- Const.;- Enhanced enumerations;- Constant expressions;- Auto;- Overview of I/O streams;- Function prototypes;- Pass-by-reference;- Default arguments;- Function and operator overloading;- String class. <p>Linkage and Storage:</p> <ul style="list-style-type: none">- Learn some tricky features often overlooked in C, but necessary for C++;- Scope;- Linkage;- Linking C and C++;- Namespace;- Static, automatic and dynamic storage;- New and delete;- Placement new;- Arrays and Pointers;

- Vectors.

C++ Development Environments for Embedded Systems:

- Compiler Optimization;
- Object files;
- Linkers and linker files;
- Loader;
- Makefiles;
- Integrated development environments;
- Debuggers;
- In-circuit emulation;
- Debug with Simulator/Emulator.

An introduction to object-oriented programming.

Classes and Objects:

- Introduction to modeling and abstraction;
- Information hiding;
- Abstract data types;
- Classes and objects;
- Public and private class members;
- Member functions;
- Scope resolution;
- This pointer.

Constructors:

- How to ensure that objects are properly initialised, and how to tidy up afterwards;
- Constructors;
- Destructors;
- Delegating constructors;
- Copy constructors;
- Pointers and objects;
- Move semantics.

Members and Friends:

- More features of C++ classes;
- Friends;
- Operator overloading;
- Overloading assignment;
- Move assignment;
- Memory fragmentation;
- Working with memory pools;
- Static members;
- Constant objects and members.

Master the subtleties of object-oriented programming in C++.

Object-Oriented Modelling and The UML:

- Learn the principles of object-oriented design;
- Class relationships;
- The Unified Modeling Language;

- Class and object diagrams;
- Association;
- Composition;
- Dependency;
- Implementing class relationships in C++;
- Initialization of class members;
- Singleton class;
- Design Patterns.

Inheritance:

- Derived classes;
- Inheritance;
- Protected members;
- Casting pointers;
- Order of initialization.

Virtual Function:

- Inheriting common behavior;
- Overriding methods;
- Virtual functions;
- Polymorphism;
- Late binding;
- Virtual destructors;
- Vtable;
- Cost of virtual functions;
- Abstract base classes and pure virtual functions;
- Interface classes.

Learn to make use of some Standard libraries Develop your own class templates.

Further C++ Features:

- User-defined conversions;
- Explicit functions;
- Defaulted and deleted behavior;
- Run-Time Type Identification;
- Type casts;
- Nested classes;
- Multiple Inheritance;
- Pointer-to-member;
- Function wrapper.

Templates:

- Function templates;
- Class templates;
- Template arguments;
- Template specialization;
- Dependent name and type lookup;
- Avoiding code bloat.

Standard Libraries:

- Summary of the Standard C and C++ libraries;
- Container classes;
- Container adapters;
- Creating and accessing containers;
- Initializer lists;
- Custom allocators;
- Std::array;
- Other C++11 Enhancements;
- Iterators;
- Range-for loop.

Understand Safety and Multi-tasking Issues.**Defensive Programming:**

- Coding standards;
- Code analysis;
- Compile-time assertions;
- Run-Time errors;
- Throwing and catching exceptions;
- Handlers;
- Standard exception classes;
- Preventing memory leakage;
- Unique_ptr.;
- Exception specification;
- Exceptions in Embedded Systems.

Principles of Real-time Operating Systems:

- Concurrency;
- Tasks and task switching;
- FreeRTOS;
- Creating and running tasks;
- Synchronisation;
- Process scheduling and pre-emption;
- Priority inversion;
- Mutex, Semaphore and Queue;
- Use of semaphores with interrupts;
- RTOS services.

Standard Library Algorithms:

- Predicates;
- Function Objects;
- Bind;
- Lambda Functions;
- Filling a container;
- Non-modifying operations;
- Transform;
- Searching;
- Sorting;
- Summary of the Standard algorithms.

	<p>C++ State Machines (Optional Topics):</p> <ul style="list-style-type: none"> - State machine representations; - Single class state machine; - State Design Pattern; - Boost mpl example. <p>Strings and Streams (Optional Topics):</p> <ul style="list-style-type: none"> - Another look at string class; - I/O stream hierarchy; - Formatted and unformatted streams; - I/O of user-defined types; - Manipulators; - File Streams; - Buffering; - Stringstreams.
Prasības kursa pasniedzējam:	<ul style="list-style-type: none"> • Pasniedzējs ir sertificēts pasniegt attiecīgo kursu. • Pasniedzēja izglītība apmācību nozarē nav zemāka par kvalifikāciju, ko iegūs nodarbinātie, apgūstot apmācību programmu.
Mācību materiāli:	<ul style="list-style-type: none"> • Drukāti vai elektroniski (ja ir paredzēti izdales mācību materiāli). • Katram kursa dalībniekam jāizsniedz 1 mācību materiāla eksemplārs un viena mācību materiāla titullapas (noformēta atbilstoši normatīvo aktu prasībām attiecībā uz vizuālo identitāti) kopija Pasūtītājam (ja ir paredzēti izdales mācību materiāli).
Mācību kursa tehniskais nodrošinājums:	Apmācību laikā nepieciešamo tehnisko nodrošinājumu nodrošina Piegādātājs.
Mācību kursa ilgums vienam dalībniekam (akadēmiskās stundas (1 māc.st. = 45 minūtes)):	40
Apmācību norises vieta:	Latvija vai Eiropas Savienība
Apmācāmo skaits:	6
Apmācāmo grupu skaits:	Pēc vajadzības
Citas prasības:	<ul style="list-style-type: none"> • Piegādātājs nodrošinās, ka tiek uzrādīts Projekta numurs "1.2.2.1/16/A/011" – mācību materiālos, sertifikātos u.tml., kā arī visas pārējās Eiropas Savienības fondu publicitātes un vizuālās identitātes prasības apmācību laikā (mācību telpās); • Apmācību veids: klātienē ar pasniedzēju; • Apmācību valoda: latviešu vai angļu • Piegādātājs nodrošinās, ka apmācību laikā tiek aizpildīts apmeklējuma reģistrācijas saraksts.

<p>Mācību kursa nosaukums:</p>	<p>Free RTOS Real-Time Programming</p>
<p>Mācību kursā obligāti ietveramas tēmas:</p>	<p>Cortex-M resources used by RTOS. Cortex-M Architecture Overview: - Two stacks pointers; - Different Running-modes and Privileged Levels; - MPU Overview; - SysTick Timer Description.</p> <p>Exception/Interrupt Mechanism Overview: - Interrupt entry and return Overview; - SVC/PendSV/Systick Interrupt Presentation;</p> <p>Developing with the IDE. Lab: Interrupt Management on Cortex-M4.</p> <p>Element of a Real-Time system. Base Real-Time concepts The Real-Time constraints Multi-Task and Real-Time; Tasks and Task Descriptors: - Content of the task descriptor; - List of task descriptors.</p> <p>Context Switch Task Scheduling and Preemption: - Tick based or tickless scheduling.</p> <p>Scheduling systems and schedulability proof: - Fixed priorities scheduling; - RMA and EDF scheduling.</p> <p>Scheduling through FreeRTOS: - Deterministic preemptive scheduling; - Scheduling strategies; - Cooperative scheduling; - Hybrid scheduling. Lab: Analyse a Context Switch.</p> <p>Task Management. The Task life-cycle: - Creating tasks; - Deleting tasks; - The Endless-loop pattern.</p> <p>Task Priorities: - Assigning task priorities; - Changing task priorities.</p>

The idle task:

- Idle task hooks.

Timing.

Lab: Managing tasks.

Resource Management.

Mutual exclusion through FreeRTOS:

- Critical sections (interrupt masking);
- Suspending (locking) the scheduler;
- Mutexes.

Mutexes concepts:

- Mutex or Semaphore;
- Recursive or not recursive mutexes;
- Priority inversion problem;
- Priority inheritance (the automatic answer);
- Priority ceiling (the design centric answer).

Gatekeeper tasks.

Lab: Implement mutual exclusion between tasks.

Synchronization Primitives.

Introduction:

- Waiting and waking up tasks;
- Semaphores;
- Events;
- Mailboxes.

Binary Semaphores through FreeRTOS:

- Give a Binary Semaphore;
- Take a binary Semaphore.

Queue Management through FreeRTOS:

- Creation;
- Sending on a queue;
- Receiving from a queue;
- Data management;
- Sending compound types;
- Transferring large data.

Event groups.

Lab: Synchronizing a task with another one through binary semaphores.

Lab: Synchronizing a task with another one through queues.

Parallelism Problems Solution.

Parallel programming problems:

- Uncontrolled parallel access;

- Deadlocks;
- Livelocks;
- Starvation.

Lab: The producer-consumer problem, illustrating (and avoiding) concurrent Access problems.

Lab: The philosopher's dinner problem, illustrating (and avoiding) deadlock, livelock and starvation.

Interrupt Management.

Need for interrupts in a Real-Time system:

- Software Interrupt;
- Time Interrupts;
- Device Interrupts.

Level or Edge interrupts.

Hardware and Software acknowledge.

Interrupt vectoring.

Interrupts and scheduling.

Deferred interrupt processing through FreeRTOS:

- Tasks with interrupt synchronization;
- Using semaphores within an ISR;
- Counting semaphores;
- Using queues within an ISR.

FreeRTOS interrupt processing:

- Writing ISRs in C;
- Interrupt safe functions;
- Interrupt nesting.

Lab: Synchronize Interrupts with tasks.

Software Timer:

- The Timer Daemon Task;
- Timer Configuration;
- One-shot/Auto-reload Timer;
- Software Timer API;
- Deferred interrupt handling.

Lab: Implement Soft Timers.

FreeRTOS-MPU.

The Cortex/M MPU:

- User and privileged modes;
- Access permissions.

Defining MPU regions:

- Overlapping regions;

	<ul style="list-style-type: none"> - Predefined regions; - Programmer-defined regions. <p>Needed linker configuration. Practical usage tips. Lab: Implement protected memory regions.</p> <p>Optional additional topics.</p> <p>Data structures. Need for specific data structures. Data structures:</p> <ul style="list-style-type: none"> - Linked lists; - Circular lists; - FIFOs; - Stacks. <p>Data structures integrity proofs:</p> <ul style="list-style-type: none"> - Assertions; - Pre and post-conditions. <p>Thread safety. Lab: Build a general purpose linked list.</p> <p>Memory management. Memory management algorithms:</p> <ul style="list-style-type: none"> - Buddy System; - Best fit/First Fit; - Pools Management. <p>FreeRTOS-provided memory allocation schemes:</p> <ul style="list-style-type: none"> - Allocate-only scheme; - Best-fit without coalescing; - Thread-safe default malloc. <p>Checking remaining free memory. Adding an application-specific memory allocator. Memory management errors. Stack monitoring. Lab: Write a simple, thread safe, buddy system memory manager. Lab: Write a generic, multi-level, memory manager. Lab: Enhance the memory manager for memory error detection.</p> <p>Trouble Shooting. Checking for stack problems. Common pitfalls.</p>
<p>Prasības kursa pasniedzējam:</p>	<ul style="list-style-type: none"> • Pasniedzējs ir sertificēts pasniegt attiecīgo kursu. • Pasniedzēja izglītība apmācību nozarē nav zemāka par

	kvalifikāciju, ko iegūs nodarbinātie, apgūstot apmācību programmu.
Mācību materiāli:	<ul style="list-style-type: none"> • Drukāti vai elektroniski (ja ir paredzēti izdales mācību materiāli). • Katram kursa dalībniekam jāizsniedz 1 mācību materiāla eksemplārs un viena mācību materiāla titullapas (noformēta atbilstoši normatīvo aktu prasībām attiecībā uz vizuālo identitāti) kopija Pasūtītājam (ja ir paredzēti izdales mācību materiāli).
Mācību kursa tehniskais nodrošinājums:	Apmācību laikā nepieciešamo tehnisko nodrošinājumu nodrošina Piegādātājs.
Mācību kursa ilgums vienam dalībniekam (akadēmiskās stundas (1 māc.st. = 45 minūtes)):	24
Apmācību norises vieta:	Latvija vai Eiropas Savienība
Apmācāmo skaits:	6
Apmācāmo grupu skaits:	Pēc vajadzības
Citas prasības:	<ul style="list-style-type: none"> • Piegādātājs nodrošinās, ka tiek uzrādīts Projekta numurs "1.2.2.1/16/A/011" – mācību materiālos, sertifikātos u.tml., kā arī visas pārējās Eiropas Savienības fondu publicitātes un vizuālās identitātes prasības apmācību laikā (mācību telpās); • Apmācību veids: klātienē ar pasniedzēju; • Apmācību valoda: latviešu vai angļu • Piegādātājs nodrošinās, ka apmācību laikā tiek aizpildīts apmeklējuma reģistrācijas saraksts.

PRETENDENTA VĒLAMIE ATLASĒS KRITĒRIJI

Apraksts	Prasības
Piedāvājuma iesniegšana	Piedāvājumu var iesniegt par visu iepirkuma apjomu kopā, iesniedzot tikai vienu piedāvājuma variantu.
Paredzamā līguma izpildes vieta	Latvija
Paredzamā iepirkuma līgumcena (EUR, bez PVN)	30 500.00 EUR
Paredzamais līguma izpildes termiņš	2018. gada 31. decembris
Apmācību veids	Klātiene ar pasniedzēju
Apmācību valoda	Latviešu vai angļu valoda
Sertificēšanas un eksaminācijas nosacījumi	Pēc apmācību kursu apguves Pretendentam ir jāizsniedz dalībniekiem sertifikāts par apmācību kursa apguvi.
Citi nosacījumi Pretendentiem	<p>1) Apmācību norises vietā jānodrošina vides un informācijas pieejamība, nepieciešamības gadījumos, nodrošinot atbilstošas palīgierīces;</p> <p>2) Jānodrošina, ka tiek uzrādīts projekta numurs „1.2.2.1/16/A/011” - mācību materiālos, sertifikātos u.tml., kā arī visas pārējās Eiropas Savienības fondu publicitātes un vizuālās identitātes prasības apmācību laikā (mācību telpās), ievērojot Ministru kabineta 2015.gada 17.februāra MK noteikumus Nr.87 „Kārtība, kādā Eiropas Savienības struktūrfondu un Kohēzijas fonda ieviešanā 2014.–2020.gada plānošanas periodā nodrošināma komunikācijas un vizuālās identitātes prasību ievērošana”.</p>
Kvalifikācijas prasības pasniedzējiem	<p>1) Pasniedzējs ir sertificēts pasniegt attiecīgo kursu.</p> <p>2) Pasniedzēja izglītība apmācību nozarē nav zemāka par kvalifikāciju, ko iegūs nodarbinātie, apgūstot apmācību programmu.</p>
Prasības pretendentiem	<p>1) Pretendents nav atzīts par maksātnespējīgu, tam nepiemīt maksātnespējas pazīmes, tas neatrodas likvidācijas stadijā, tā saimnieciskā darbība nav apturēta vai pārtraukta, nav uzsākta tiesvedība par pretendenta darbības izbeigšanu, maksātnespēju vai bankrotu.</p> <p><i>Latvijā reģistrētam uzņēmumam Pasūtītājs minēto informāciju pārbauda pēc publiskajās datu bāzēs pieejamajiem datiem. Ārvalstīs reģistrētam uzņēmumam jāiesniedz apliecinājums par atbilstību kritērijiem.</i></p> <p>2) Pretendentam nav nodokļu vai citu valsts obligāto maksājumu parādi Latvijā vai valstī, kurā tas reģistrēts.</p> <p><i>Latvijā reģistrētam uzņēmumam Pasūtītājs minēto informāciju pārbauda pēc publiskajās datu bāzēs pieejamajiem datiem. Ārvalstīs reģistrētam uzņēmumam jāiesniedz apliecinājums par atbilstību kritērijiem.</i></p> <p>3) Pretendentam ir jānodrošina starptautiski atzīti dalībnieku sertifikācijas eksāmeni. Tehniskajā piedāvājumā Pretendentam ir jānorāda sertifikācijas iestāde, saskaņā, ar kuru tiks nodrošināts eksāmens un izdots sertifikāts.</p>
Prasības piedāvājuma noformējumam	Pretendents līdz 2018.gada 29. jūnija plkst. 16:00

	<p>iesniedz 1 (vienu) piedāvājuma oriģinālu latviešu vai angļu valodā datorrakstā aizlīmētā un apzīmogatā aploksnē vai paketē, uz kuras norādīta šāda informācija:</p> <ul style="list-style-type: none">• Pasūtītāja adrese;• Pretendenta nosaukums un juridiskā adrese;• Norāde: Iepirkuma procedūrai "Iegultās elektronikas programmēšanas kompetences paaugstināšanas apmācības" (Iepirkuma identifikācijas Nr. 6-8/A-29) Neatvērt līdz 2018.gada 29.jūnija plkst. 16.00.
--	---

- 1) Pretendentu piedāvājumu vērtēšanu nodrošina Pasūtītāja izveidota iepirkuma komisija slēgtā sēdē.
- 2) Iesniedzot piedāvājumu, Pretendents pilnībā pieņem iepirkuma nolikumā un tehniskajā specifikācijā ietvertos noteikumus. Jebkura Pretendenta piedāvātā norma, kas ir pretrunā ar iepirkuma prasībām, var būt par iemeslu piedāvājuma noraidīšanai.
- 3) Gadījumā, ja Pretendents iesniedzis nepamatoti lētu piedāvājumu, komisija pieprasa sniegt rakstisku paskaidrojumu un dokumentāli pierādīt zemās cenas veidošanās pamatotību, jo Pasūtītājs vēlas saņemt kvalitatīvus apmācību pakalpojumus atbilstoši Eiropas Savienības fondu programmas noteikumiem. Ja pretendents 5 (piecu) darba dienu laikā pēc pieprasījuma saņemšanas nespēj dokumentāli pierādīt zemās cenas veidošanos, komisija pieņem lēmumu par pretendenta izslēgšanu no turpmākās dalības iepirkuma procedūrā.
- 4) Pasūtītājs var pieprasīt no Piegādātāja iesniegtā Piedāvājuma precizējošu informāciju (mācību kursu saturu, mācību materiālu saturu, pasniedzēju kvalifikāciju un pieredzi, pasniedzēju kontaktinformāciju (kas var tikt izmantota, lai pārliecinātos par pasniedzēju gatavību piedalīties pakalpojuma sniegšanā)) un Piegādātāju, vai citu informāciju, kas saistīta ar pienācīgu Piegādātāja līgumsaistību izpildes iespēju pārbaudi. Precizējošā informācija Piegādātājam ir jāiesniedz 5 (piecu) darba dienu laikā no Pasūtītāja pieprasījuma nosūtīšanas dienas. Ja Piegādātājs noteiktajā termiņā nav iesniedzis pieprasīto papildus informāciju, Pasūtītājs ir tiesīgs noraidīt iesniegto piedāvājumu, uzskatot, ka Piegādātājs neuztur spēkā savu piedāvājumu.

Pretendents, iesniedzot piedāvājumu, izmanto piedāvājuma formas:

„Pretendenta pieteikums” (Pielikums Nr.1);

„Pretendenta tehniskais piedāvājums” (Pielikums Nr.2);

„Pretendenta finanšu piedāvājums” (Pielikums Nr.3).

PRETENDENTA PIETEIKUMS:SĀKOTNĒJAIS PIEDĀVĀJUMS GALĪGAIS PIEDĀVĀJUMS *(Pretendents atzīmē atbilstošo piedāvājuma veidu. Ja Pretendents neatzīmē nevienu no piedāvājuma veidiem, Pasūtītājs uzskata, ka izteikts galīgais piedāvājums)***1. IESNIEDZĒJS**

Nosaukums	
Reģistrācijas Nr.	
Juridiskā adrese	
Faktiskā adrese	
Tālrunis	

2. IESNIEDZĒJA KONTAKTPERSONA

Vārds, uzvārds	
Adrese	
Tālrunis	
E-pasta adrese	

Pretendents ar šī pieteikuma iesniegšanu:

- 1) piesakās piedalīties iepirkuma procedūrā „legultās elektronikas programmēšanas kompetences paaugstināšanas apmācības” (iepirkuma identifikācijas Nr. 6-8/A-29);
- 2) apliecina, ka ir iepazinies ar iepirkuma procedūras nolikumu un apņemas ievērot tā prasības;
- 3) apņemas pasūtījuma piešķiršanas gadījumā slēgt Iepirkuma līgumu ar Pasūtītāju;
- 4) atzīst sava piedāvājuma spēkā esamību ne īsāku kā līdz 2018.gada 31.decembrim no piedāvājumu atvēršanas sēdes dienas.
- 5) garantē, ka visas sniegtās ziņas ir patiesas.

Vārds, uzvārds	
Amats	
Paraksts	
Datums	

Pielikums Nr.2

Iepirkuma procedūra "Iegultās elektronikas programmēšanas kompetences paaugstināšanas
apmācības"

(Iepirkuma identifikācijas Nr. 6-8/A-29)

PRETENDENTA TEHNISKAIS PIEDĀVĀJUMS:

Mācību kurs Nr. _____

Mācību kursa nosaukums: _____

Mācību kursa saturs:	
Mācību kursa pasniedzējs (vārds, uzvārds): (Piedāvājuma pielikumā jāpievieno CV)	
Prasības kursa pasniedzējam: (Piedāvājuma pielikumā jāpievieno sertifikāta kopija)	
Sertifikācijas standarta un sertifikācijas centra, kur tiks nodrošināts eksāmens, nosaukums:	
Mācību materiāli:	
Mācību kursa tehniskais nodrošinājums:	
Mācību kursa ilgums vienam dalībniekam (akadēmiskās stundas):	
Apmācību vieta:	
Apmācāmo skaits:	
Apmācību veids:	
Apmācību valoda:	

_____ (vārds, uzvārds)

(paraksts)

z.v.

PRETENDENTA FINANŠU PIEDĀVĀJUMS:**1. Kopējā izmaksu tāme:**

Mācību kursa Nr. _____

Mācību kursa nosaukums: _____

Izmaksu pozīcija	Vienība	Vienību skaits	Vienības izmaksas bez PVN (EUR)	Izmaksas bez PVN (EUR)
1. Pasniedzēju darba atlīdzība par darba stundām (tai skaitā darba devēja valsts sociālās apdrošināšanas obligātās iemaksas)	stundas (akadēmiskās)			
2. Mācību materiālu izmaksas (ierobežojumu skatīt šī nolikuma punktā Nr.2.1.)	gab.	1		
3. Apmācībai izmantojamo telpu nomas izmaksas par mācību laiku	stundas (akadēmiskās)			
4. Apmācībai izmantojamo iekārtu, instrumentu vai aprīkojuma nomas izmaksas par mācību laiku	stundas (akadēmiskās)			
5. Apmācāmo sertifikācijas un eksaminācijas izmaksas	dalībnieki	1		
6. Izmaksas apmācāmo darbinieku apmācību vajadzību noteikšanai, karjeras konsultācijām un zināšanu līmeņa testēšanai	dalībnieki	1		
Mācību kursa cena 1 (vienam) dalībniekam bez PVN (EUR):				
Mācību kursa izmaksas kopā kursa kopējam plānotajam dalībnieku skaitam bez PVN (EUR):				

_____ (vārds, uzvārds)

(paraksts)

z.v.

2.1. Apmācībām izmantojamie mācību materiāli pēc apmācību beigām paliek projektā apmācīto dalībnieku īpašumā. Apmācībām izmantojamo mācību materiālu izmaksu summa nedrīkst pārsniegt 50 EUR uz vienu apmācāmo. Ja gadījumā izmaksas pārsniedz noteikto summu, Pretendentam kopā ar piedāvājumu jāiesniedz izmaksu pamatojums, lai Pasūtītājs pamatojumu par lietderību pēc tam var iesniegt projekta sadarbības iestādē.

2.2. Pretendents apmācību kursa cenu nosaka par 1 (vienu) dalībnieku bez PVN, saskaņā ar Finanšu piedāvājuma formu.

2.3. Nosakot līgumcenu par iepirkuma priekšmeta 1 (vienu) mācību kursu, Pasūtītājs izmanto šādu formulu: piedāvātā mācību kursa cena 1 (vienam) dalībniekam x kopējais kursā plānotais dalībnieku skaits.

2.4. Pasūtītājs vienojoties ar Pretendentu, līguma darbības laikā atsevišķos mācībuursos var pasūtīt apmācības gan mazākam, gan lielākam personu skaitam, saglabājot nemainīgas vienas vienības izmaksas. Pasūtītājam ir tiesības līguma darbības laikā palielināt apmācāmo skaitu un kopējo līgumcenu, ja līguma darbības laikā pēc tehniskajā specifikācijā norādītajiem kursiem ir radies papildus pieprasījums no projektā iesaistīto komersantu puses, kas Pasūtītājam uz iepirkuma veikšanas brīdi nebija zināms un radies līguma darbības laikā.

Ja iepirkuma līguma darbības laikā atsevišķos mācībuursos tiek apmācīts lielāks personu skaits kā norādīts tehniskajā specifikācijā, Pretendenta piedāvājumā noteiktā kursu cena 1 (vienam) dalībniekam nevar tikt mainīta.

2.5. Pasūtītājs līguma darbības laikā var neizmantot kādu konkrētu mācību kursu, ja iepirkuma līguma izpildes laikā Pasūtītājam pēc tā nav pieprasījuma no projektā iesaistīto komersantu puses.

2.6. Mainoties mācību kursu versijai, Pretendents, iepriekš saskaņojot ar Pasūtītāju, var Tehniskajā specifikācijā iekļauto apmācību kursu aizstāt ar tā jaunāku versiju, būtiski nemainot Tehniskajā piedāvājumā sniegto kursa saturu un nemainot Finanšu piedāvājumā minēto apmācību kursa summu.

_____ (vārds, uzvārds)
(paraksts)

Z.V.