**Educational profiles**

Program has one educational profile. Mobile application profile is focused on the development of android Mobile alications for educational and medical applications.

**Content of the program**

Mathematics for Mobile Development:

| Courses | | | Author |
|---|---|---|---|
| Code | Title | Volume (ECTS) | |
| | Obligatory Courses | 15 | |
| 1.1 | Machine Learning | 5 | Veriga A.V. |
| 1.2 | Computer Vision | 5 | Veriga A.V. |
| 1.3 | Artificial Intelligence | 5 | Koshkarov A.V. |

Programming for Mobile Development:

| Courses | | | Author |
|---|---|---|---|
| Code | Title | Volume (ECTS) | |
| | Obligatory Courses | 15 | |
| 2.1 | Java Programming | 5 | Podgorniy A.V. |
| 2.2 | Cross-platform Development | 5 | Morozov B.B. Evdoshenko O.I. |
| 2.3 | Algorithms and data structures | 5 | Podgorniy A.V. |

Interdisciplinary:

| Courses | | | Author |
|---|---|---|---|
| Code | Title | Volume (ECTS) | |
| | Obligatory Courses | 15 | |
| 3.1 | Database | 5 | Evdoshenko O.I. |
| 3.2 | WEB Programming | 5 | Morozov B.B. |
| 3.3 | Big Data Analytics | 5 | Koshkarov A.V. |

Mobile applications design:

| Courses | | | Author |
|---|---|---|---|
| Code | Title | Volume (ECTS) | |
| | Obligatory Courses | 15 | |
| 4.1 | Software engineering | 4 | Evdoshenko O.I. |
| 4.2 | UI/UX design | 4 | Evdoshenko O.I. |
| 4.3 | Economics and project management | 4 | |
| 4.4 | Psychology of personal and team efficiency | 3 | |

Professional module: Mobile Development

| Courses | | | Author |
|---|---|---|---|
| Code | Title | Volume (ECTS) | |
| | Obligatory Courses | 24 | |
| 5.1 | Specifics of Android Development | 5 | Koshkarov A.V. |
| 5.2 | Neural Networks for Mobile Applications | 5 | Veriga A.V. |
| 5.3 | Mobile Applications Security | 5 | Podgorniy A.V. |
| 5.4 | AR and VR Technologies for Mobile Applications | 5 | |
| 5.5 | Cloud services for mobile development | 4 | Morozov B.B. |

| Subject code: | Subject name: **Machine Learning** | | |
|---|---|---|---|
| **Study load:** 5 ECTS | **Load of contact hours:** | **Study semester:** | **Assessment:** |
| **Objectives:** | The goal of this course is to develop students ' expertise in major areas of Machine Learning including Prediction, Classification, Clustering, and Information Retrieval; in analysis of large and complex datasets, creating systems that adapt and improve over time, and building intelligent applications that can make predictions from data. | | |
| **Course outline:** | The course is offered as a combination of theoretical (lectures) and practical (seminars/workshops) parts. Students design their own models of Neural Network and improve them from simple to complex during the course. After the completion of each thematic section (chapter), they are to apply their new knowledge to the model and thus see the improvement in practice. Using some datasets from real-world enables students to resolve some actual problems by researching the state-of-the-art solutions in the field of machine learning. | | |
| **Learning Outcomes:** | On completion of the course, students are expected to be able to: 1. Identify potential applications of machine learning in practice. 2. Describe the core differences in analyses enabled by regression, classification, and clustering. 3. Apply regression, classification, clustering, retrieval, recommender systems, and deep learning. 4. Represent your data as features to serve as input to machine learning models. 5. Assess the model quality in terms of relevant error metrics for each task. 6. Utilize a dataset to fit a model to analyze new data. | | |
| **Assessment Methods:** | Skills will be assessed using a variety of strategies, including: 1. Tests (open-answer or multiple choice) or quizzes to assess "knowing" and "remembering" skills. 2. Group/individual projects, portfolios, case study, role playing 3. Self- and peer-assessment, feedback (focus is on comments rather than grades) 4. Labworks | | |
| **Teacher(s):** | Veriga A.V. | | |
| **Prerequisite subject(s):** | Linear Algebra Multivariable Calculus Python programming | | |
| **Compulsory Literature:** | Bishop, Pattern Recognition and Machine Learning, 2006 Duda, Hart, and Stork, Pattern Classification, 2nd Ed., 2002 | | |
| **Replacement Literature:** | Ripley, Pattern Recognition and Neural Networks, 1996 Tan, Steinbach, and Kumar, Introduction to Data Mining, Addison-Wesley, 2005. Hastie, Friedman, and Tibshirani, The Elements of Statistical Learning, 2001 | | |
| **Participation and** | Classroom attendance and active participation is important because most of | | |

| | |
|---|---|
| **Exam requirements:** | the topics have tasks which require team work and imply workshops. Some experiments with model tuning are long term and most optimal on a few simultaneous starts of learning process with different hyper-parameters and succeeding discussion. |
| **Independent work:** | Normal equation of a line (additional topic)<br>Housing Price Forecasting (practices)<br>Handwriting recognition (practices)<br>Complex Image Classifier (practices)<br>Classification of cats and dogs on 25 thousand images (practices)<br>Organizing the text of BBC News (practices)<br>Character Generation Job Using Character-Based RNN (practices)<br>The problem of exploding and disappearing gradients. (additional topic) |
| **1) Date 1** | Linear Algebra. Vectors and Matrices |
| **2) Date 2** | Multivariable linear regression |
| **3) Date 3** | Logistic regression. Classification and representation |
| **4) Date 4** | Logistic Regression Model |
| **5) Date 5** | Introduction in Machine Learning. New paradigm of programming |
| **6) Date 6** | Google Collaboratory and Tensorflow, crash course |
| **7) Date 7** | Computer vision |
| **Date 8** | Convolution Neural Networks |
| **Date 9** | Explore of big datasets |
| **Date 10** | Augmentation and transferring |
| **Date 11** | Practical aspects of deep learning. Overfitting, hyperparameter tuning. |
| **Date 12** | Regularization |
| **Date 13** | Natural Language Processing. Word embeddings |
| **Date 14** | Sequences, Time Series and Prediction |
| **Date 15** | Generative-adversarial Neural Networks |
| **Date 16** | Reinforcement learning |

| | | | |
|---|---|---|---|
| **Subject code:** | Subject name: **Computer Vision** | | |
| **Study load:**<br>5 ECTS | **Load of contact hours:** | **Study semester:** | **Assessment:** |
| **Objectives:** | The goal of this course is to implement the foundational layers of CNNs (pooling, convolutions) and to stack them properly in a deep network to solve multi-class image classification problems, to get acquainted with techniques and methods used in deep CNNs, to learn the concepts of object detection, art generation and face recognition. | | |
| **Course outline:** | The course is offered as a combination of theoretical (lectures) and practical (seminars/workshops) parts. After each module completed, students have to pass a test and one or two programming assignments to apply their new knowledge in practice. Using datasets from real-world enables students to resolve some actual problems by researching the state-of-the-art solutions in the field of Computer vision. | | |
| **Learning Outcomes:** | On completion of the course, students are expected to be able to:<br>1. Build a convolutional neural network, including recent variations such as residual networks.<br>2. Apply convolutional networks to visual detection and recognition tasks. | | |

| | |
|---|---|
| | 3. Use neural style transfer to generate art.<br>4. Apply these algorithms to a variety of image, video, and other 2D or 3D data. |
| **Assessment Methods:** | Skills will be assessed using a variety of strategies, including:<br><br>1. Tests (open-answer or multiple choice) or quizzes to assess "knowing" and "remembering" skills.<br>2. Group/individual projects, portfolios, case study, role playing<br>3. Self- and peer-assessment, feedback (focus is on comments rather than grades)<br>4. Labworks |
| **Teacher(s):** | Veriga A.V. |
| **Prerequisite subject(s):** | Linear Algebra<br>Multivariable Calculus<br>Python programming |
| **Compulsory Literature:** | Shapiro, L., and Stockman, G., Computer Vision, Upper Saddle River, NJ: Prentice Hall, 2001. |
| **Replacement Literature:** | Bernd Jähne and Horst Haußecker (2000). Computer Vision and Applications, A Guide for Students and Practitioners. Academic Press. ISBN 0-12-379777-2.<br>David A. Forsyth and Jean Ponce (2003). Computer Vision, A Modern Approach. Prentice Hall.ISBN 0-13-085198-1. |
| **Participation and Exam requirements:** | Classroom attendance and active participation is important because most of the topics have tasks which require team work and imply workshops. Some experiments with model tuning are long term and most optimal on a few simultaneous starts of learning process with different hyper-parameters and succeeding discussion.<br>All the tests are held online so they can be passed out of the class. |
| Independent work: | 1. Keras tutorial (self study)<br>2. Convolutional Model (programming);<br>3. Residual Network (programming);<br>4. Car detection with YOLO (programming);<br>5. Face recognition & Neural style transfer (programming); |
| **Date 1** | Foundations of Convolutional Neural Networks. edge detection, padding, **strided** convolutions. |
| **Date 2** | Convolutions over volume. Pooling layer. Simple CNN example |
| **Date 3** | Deep convolutional models. Dimensionality reduction of a volume in a very deep network |
| **Date 4** | Data augmentation. Transfer Learning. Residual Networks. Inception Network. |
| **Date 5** | Object Localization. Landmark Detection. Object Detection. Bounding Box prediction |
| **Date 6** | Intersection Over Union. Non-max Suppression. Anchor Boxes |
| **Date 7** | YOLO Algorithms. Car detection. Region Proposals. |
| **Date 8** | Face recognition. One Shot Learning. Siamese Network |
| **Date 9** | Triplet Loss. Face verification and binary classification |
| **Date 10** | Neural style transfer. Content cost function. Style cost function. 1D and 3D generalization |
| **Date 11** | Matrix capsules with EM routing. Expectation-Maximization procedure. Using EM for routing-by-agreement |

| Date 12 | Computational Mirrors. Blind Inverse Light Transport by Deep Matrix Factorization |
|---|---|
| Date 13 | Emotion Recognition, Gesture Recognition, Action Detection, Video Object Segmentation. |
| Date 14 | Medical Image Segmentation. |

| Subject code: | Subject name: Artificial Intelligence | | |
|---|---|---|---|
| Study load:<br>5 ECTS | Load of contact hours: | Study semester: | Assessment: |
| Objectives: | The goal of this course is to provide an overall understanding of artificial intelligence technologies and their applications. | | |
| Course outline: | The course focuses on the main fields of artificial intelligence that are currently popular. It includes hands-on work on creating intelligent systems using classic machine learning and deep learning methods.<br><br>Topics covered:<br>1. Classical machine learning<br>2. Computer vision<br>3. Natural language processing<br>4. Reinforcement training<br>5. Generative machine learning<br><br>Special attention is given to annotating data and preparing data for use in models. Additional materials will be provided through the online platform. | | |
| Learning Outcomes: | On completion of the course, students are expected to be able to:<br>1. Know objectives and functions of modern AI.<br>2. Detect of an AI problem based on its characteristics and its constraints.<br>3. Understand and implement search and adversarial (game) algorithms.<br>4. Understand machine learning algorithms and extract knowledge models from data. | | |
| Assessment Methods: | Skills will be assessed using a variety of strategies, including:<br><br>1. Tests (open-answer or multiple choice) or quizzes to assess "knowing" and "remembering" skills.<br>2. Group/individual projects, portfolios, case study, role playing<br>3. Self- and peer-assessment, feedback (focus is on comments rather than grades)<br>4. Labworks<br><br>Laboratory work is carried out in mini-groups, thus helping to develop teamwork and complex problems solving skills. Students are to prepare an electronic report on the results of each laboratory work. | | |
| Teacher(s): | Koshkarov A.V. | | |
| Prerequisite subject(s): | None | | |

| Compulsory Literature: | • Goodfellow I., Bengio Y., Courville A. Deep learning. – MIT press, 2017.<br>• François C. Deep learning with Python. – 2017.<br>• Trask A. Grokking deep learning. – Manning Publications Co., 2019. |
|---|---|
| Replacement Literature: | • Rao D., McMahan B. Natural language processing with PyTorch: build intelligent language applications using deep learning. – " O'Reilly Media, Inc.", 2019.<br>• Raschka S., Mirjalili V. Python Machine Learning: Machine Learning and Deep Learning with Python, scikit-learn, and TensorFlow 2. – Packt Publishing Ltd, 2019. |
| Participation and Exam requirements: | Classroom attendance and active participation is important because most of the topics have tasks which require team work and imply workshops. In addition, a presentation of team projects is expected, final assessment will be based on the results of it. |
| Independent work: | Additional work includes a deeper study of machine learning algorithms (including their mathematical foundations). This will help students learn more about artificial intelligence algorithms, increasing accuracy and quality.<br><br>Labs also provide a deeper understanding of deep learning frameworks and Python libraries. |
| Date 1 | **Introduction**<br><br>Practice-oriented lecture "Introduction to artificial intelligence"<br>Lecture "Artificial Intelligence Technologies" |
| Date 2 | **Classic Machine Learning**<br><br>Lecture "Machine Learning Algorithms"<br>Lab "Supervised machine learning" |
| Date 3 | Lab "Unsupervised machine learning" |
| Date 4 | Lecture "Artificial Neural Networks"<br>Lab "Creation and training of a neural network" |
| Date 5 | Lab "Convolutional neural networks" |
| Date 6 | Lecture "Computer Vision"<br>Lab "Image labelling" |
| Date 7 | Lab "Identification of objects in the image" |
| Date 8 | Lab "Segmentation of objects in the image" |
| Date 9 | **Natural Language Processing**<br><br>Lecture "Natural Language Processing"<br>Lab "Preparing a dataset" |
| Date 10 | Lab "Creating a question-answer intellectual system" |
| Date 11 | **Reinforcement learning**<br><br>Lab "Reinforcement learning" |
| Date 12 | **Generative machine learning**<br><br>Lab "Generative machine learning" |

| Subject code: | Subject name: Java programming |
|---|---|

| Study load: 5 ECTS | Load of contact hours: | Study semester: | Assessment: |
|---|---|---|---|
| **Objectives:** | The goal of this course is to study a modern approach to programming based on object-oriented technology, to acquire skills in software development in the Java language. | | |
| **Course outline:** | The course consists of a theoretical and practical part: during the course, students learn the theory, advanced techniques of creating programs with Java and simultaneously under the guidance of a teacher develop a real-life full-fledged Java project, that could solve some actual industry problems, gaining useful experience in applying complex object-oriented concepts specific to the language. Topics covered: 1. Develop GUI applications with the Swing and JavaFX libraries 2. Advanced object oriented programming 3. Correctly handle errors, work with exceptions 4. Work with the network 5. Threading in Java | | |
| **Learning Outcomes:** | On completion of the course, students are expected to be able to: 1. Know the Java language 2. Understand basics of programming with a modern programming language, Java. 3. Understand installation and usage of good Java development environment. 4. Produce robust programs in Java using exception handling and extensive program testing. 5. Manage errors appropriately, work with the network and with exceptions. | | |
| **Assessment Methods:** | Skills will be assessed using a variety of strategies, including: 1. Tests (open-answer or multiple choice) or quizzes to assess "knowing" and "remembering" skills. 2. Group/individual projects, portfolios, case study, role playing 3. Self- and peer-assessment, feedback (focus is on comments rather than grades) 4. Labworks | | |
| **Teacher(s):** | Podgorniy A.V. | | |
| **Prerequisite subject(s):** | None | | |
| **Compulsory Literature:** | 1. Sau Prakashani. Java Programming: Basic Java programming (2019) 2. M. T. Somashekara, D. S. Guru, K. S. Manjunatha. Object | | |

| | oriented programming with java (2017) |
|---|---|
| **Replacement Literature:** | 1. John Horton. Android Programming for Beginners - Second Edition (2018) <br> 2. Neil Smyth. Android Studio 3.5 Development Essentials (2019) |
| **Participation and Exam requirements:** | Visiting the classes and active participation is important because most of the classes will be dedicated for team works and workshops and not for presentations. |
| **Independent work:** | The students prepare presentations for different themes regarding subject during independent work. |
| **1) Date 1** | Classroom presentation – GUI applications development <br><br> Practice – Develop GUI applications with the Swing library |
| **2) Date 2** | Classroom presentation – GUI applications development <br><br> Practice – Develop GUI applications with the JavaFX library |
| **3) Date 3** | Classroom presentation – Advanced object-oriented programming <br><br> Practice – Abstract classes and interfaces in Java |
| **4) Date 4** | Classroom presentation – The concept of exception handling, its comparison with the traditional error handling mechanism <br><br> Practice – Creating an application that can handle various exception types |
| **5) Date 5** | Classroom presentation – Networking in Java <br><br> Practice – Sockets; Client-Server architecture |
| **6) Date 6** | Classroom presentation – Threading in Java <br><br> Practice – Multithreaded application |

| **Subject code:** | **Subject name:** Cross-platform Development | | |
|---|---|---|---|
| **Study load:** <br> 5 ECTS | **Load of contact hours:** | **Study semester:** | **Assessment:** |
| **Objectives:** | The goal of the course is to master basic knowledge, concepts, and skills in cross-platform application development. | | |
| **Course outline:** | The goals of the course are achieved by a combination of teaching and working methods, including lectures, workshops and laboratory work performed at the computers mostly in teams. <br><br> Topics covered: <br> The concept of cross-platform application. Overview of the main hardware and software platforms. Analysis of the demand for cross-platform applications. The main operating systems. Overview of programming languages. Overview of | | |

| | |
|---|---|
| | cross-platform libraries. Overview of cross-platform execution frameworks. Choosing a programming language and tools for developing a cross-platform application. Overview of programming language features. Overview of the performance framework. Drawing analogies with already studied programming languages. Application Development for Available Systems: Linux, Windows Additional training materials are distributed through the online training platform. |
| **Learning Outcomes:** | On completion of the course, students are expected to be able to: <br><br> 1. Create mobile applications targeting multiple platforms with a single codebase; <br> 2. Use special software development environments; <br> 3. Identify erroneous and emergency situations and to respond to them, <br> 4. Apply various features of the open source framework to build truly cross-platform mobile applications. |
| **Assessment Methods:** | Skills will be assessed using a variety of strategies, including: <br><br> 1. Tests (open-answer or multiple choice) or quizzes to assess "knowing" and "remembering" skills. <br> 2. Group/individual projects, portfolios, case study, role playing <br> 3. Self- and peer-assessment, feedback (focus is on comments rather than grades) <br> 4. Labworks |
| **Teacher(s):** | Morozov B.B. <br> Evdoshenko O.I. |
| **Prerequisite subject(s):** | None |
| **Compulsory Literature:** | 1.Gamma, Erich ; Helm, Richard ; Johnson, Ralph ; Vlissides, John M.: Design Patterns: Elements of Reusable Object-Oriented Software. 1 : Addison-Wesley Professional, 1994. - ISBN 0201633612 <br> 2. Object-oriented design techniques. Design Patterns / Gamma E, Helm R, Johnson R, Vlissides D; A. Slinkin. St. Petersburg. : Peter, 2013. - 368 s. : il. <br> 3. Orlov S.A. Theory and practice of programming languages : (for bachelors and masters) : textbook on the set. "Informatics and Computing" / Orlov S.A. - St. Petersburg. : Peter, 2013. - 688 s. : il. <br> 4. Buynachev, S.K. Python Programming Basics: Textbook / S.K. Buynachev, N.Y. Boclag ; Ministry of Education and Science of the Russian Federation, Ural Federal University named after the first President of Russia B. N. Yeltsin. - Yekaterinburg : Ural University Publishing House, 2014. - 92 s. : table, il. The Bibliog. in the n. - ISBN 978-5-7996-1198-9 ; It's the same. - URL: http://biblioclub.ru/index.php?page=book&id=275962 |
| **Replacement Literature:** | 1.Making Games with Python and Pygame by Al Sweigart eBook Details:2012 ISBN/ASIN: 1469901730 ISBN-13: 9781469901732 <br> 2.Pavlovskaya T.A. C. Programming in a high-level language: "textbook for students, training on the course of the fit. "Computer science and computing" / Pavlovskaya T.A. - St. Petersburg. It's not a good time. : Peter, 2012. - 432 s. : il. <br> 3. Khakhaev, I.A. Workshop on algorithmization and programming on Python : course / I.A. Khakhaev. - 2nd year, correct. - M. : National Open University "INTUIT", 2016. - 179 s. : il. The Bibliog. in the n. ; It's the same. - URL: http://biblioclub.ru/index.php?page=book&id=429256 |

| | 4. Sweigart, A. Development of computer games in the language of Python / A. Sweigart. 2nd year, - M. : National Open University "INTUIT", 2016. - 505 s. : il. ; It's the same. - URL: http://biblioclub.ru/index.php?page=book&id=429009 |
|---|---|
| **Participation and Exam requirements:** | Visiting the classes and active participation is important because most of the classes will be dedicated for team works and workshops and not for presentations. |
| **Independent work:** | The students prepare presentations for different themes regarding subject during independent work. |
| **1) Date 1** | Main hardware and software platforms<br> Laboratory work 1 Development of a cross-platform application in Python |
| **2) Date 2** | Programming languages<br>Laboratory work 2 Development of cross-platform application in C # |
| **3) Date 3** | Cross-platform libraries<br>Laboratory work 3: Creating a cross-platform training application in Eclipse |
| **4) Date 4** | Cross-platform execution frameworks<br>Laboratory work 4 Debugging a cross-platform training application using virtual emulators |
| **5) Date 5** | Programming language features |
| **6) Date 6** | Basic techniques and methods for programming mobile apps using Python, C# |
| **7) Date 7** | Cross-platform training application in Eclipse |
| **8) Date 8** | Main hardware and software platforms<br> Laboratory work 1 Development of a cross-platform application in Python |

| **Subject code:** | **Subject name:** Algorithms and data structures | | |
|---|---|---|---|
| **Study load:**<br>5 ECTS | **Load of contact hours:** | **Study semester:** | **Assessment:** |
| **Objectives:** | The goal of the course is to study the basic data structures and algorithms that will serve as the foundation for all further knowledge in the field of computer science and software engineering. | | |
| **Course outline:** | Algorithms and data structures are the basis of every software system: distributed system, mobile applications, databases, web applications.<br><br>The course consists of theoretical and practical parts.<br>In the theoretical part students will study issues related to the development of algorithms, an assessment of their complexity, the ability to apply them in various tasks.<br>During the practical part students will learn how to implement algorithms, design data structures and build relationships between them.<br><br>Topics covered:<br>Introduction: Fibonacci numbers, computational complexity of algorithms, sort | | |

| | |
|---|---|
| | and search<br>«Divide and conquer» method: multiplication of numbers with the Karatsuba method, recursion, matrix multiplication<br>Data structures: array, heap, binary tree<br>Sort: simplest algorithms, heap sort, merge sort, quick sort<br>Graph theory: graphs and methods for their storage, depth-first search in directed and undirected graph<br>Paths in graphs: breadth-first search, Dijkstra algorithm |
| **Learning Outcomes:** | On completion of the course, students are expected to be able to:<br><br>1. Understand fundamental Data Structures<br>2. Understand fundamental abstract data types<br>3. Program data structures and use them in implementations of abstract data types.<br>4. Understand basic algorithmic complexity.<br>5. Estimate the algorithmic complexity of simple, non-recursive programs<br>6. Select appropriate data structures and algorithms for problems and to justify that choice. |
| **Assessment Methods:** | Skills will be assessed using a variety of strategies, including:<br>1.     Tests (open-answer or multiple choice) or quizzes to assess "knowing" and "remembering" skills.<br>2.     Group/individual projects, portfolios, case study, role playing<br>3.     Self- and peer-assessment, feedback (focus is on comments rather than grades)<br>4.     Labworks<br>5.     Exam. |
| **Teacher(s):** | Podgorniy A.V. |
| **Prerequisite subject(s):** | None |
| **Compulsory Literature:** | 1. Donald Ervin Knuth. The Art of Computer Programming: Fundamental Algorithms<br>2. Donald Ervin Knuth. The Art of Computer Programming: Volume 3: Sorting and Searching |
| **Replacement Literature:** | K. Thulasiraman, M. N. S. Swamy. Graphs: Theory and Algorithms |
| **Participation and Exam requirements:** | Visiting the classes and active participation is important because the classes will be mostly dedicated for team works and workshops and not for presentations. Students will earn points for every practical assignment. In the end of the study all the points will be summed up. |
| **Independent work:** | The students prepare presentations on different topics related to the course. |
| **1) Date 1** | Classroom presentation – Introduction to the course<br>Classroom presentation – Fibonacci numbers<br>Practice – Creating a program for the Fibonacci numbers calculating |
| **2) Date 2** | Classroom presentation – Computational complexity of algorithms<br>Classroom presentation – Sort and search |

| | Practice – bubble sort |
|---|---|
| **3) Date 3** | Practice – Search in array; find Max and Min; other array operations |
| **4) Date 4** | Classroom presentation – «Divide and conquer» method |
| | Practice – Matrix multiplication |
| **5) Date** | Classroom presentation – Recursion |
| | Practice – Multiplication of numbers with the Karatsuba method |
| **6) Date** | Classroom presentation – Data structures: array, heap, binary tree |
| | Practice – Heap sort, Merge sort |
| **7) Date** | Practice – Quick sort |
| **8) Date** | Classroom presentation – Introduction to the graph theory |
| | Practice – Graph storage |
| **9) Date** | Classroom presentation – DFS |
| | Practice – Depth-first search in directed and undirected graph |
| **10) Date** | Classroom presentation – Paths in graphs |
| | Practice – BFS |
| **11) Date** | Practice – Dijkstra algorithm |

Database

| **Subject code:** | **Subject name:** Database | | |
|---|---|---|---|
| **Study load:** 5 ECTS | **Load of contact hours:** | **Study semester:** | **Assessment:** |
| **Objectives:** | The objective of the course is to develop students' practical skills in database design, development, and management. | | |
| **Course outline:** | The course provides students with theoretical knowledge and practical skills in database design, development and maintenance. Students acquire knowledge about the main stages of database design, data models, principles of relationship normalization, relational algebra, internal organization of a relational database management system, gain practical skills in using the declarative SQL language for creating, modifying and managing data in a relational database, skills in working with relational database management systems: PostgreSQL, Microsoft SQL Server. Topics covered: 1. Introduction to database theory 2. Data model. Relational model 3. Relational algebra operations 4. Normalization of relations in the database 5. Introduction to MS SQL Server 6. The basics of T-SQL. DDL 7. Built-in functions. Variables and control constructs | | |
| **Learning Outcomes:** | On completion of the course, students are expected to be able to: 1. Understand database concepts and structures and query language 2. Understand Functional Dependency and Functional Decomposition. 3. Explain the basic concepts of relational data model, entity-relationship model, relational database design, relational algebra and SQL. | | |

| | |
|---|---|
| | 4. Understand query processing and techniques involved in query optimization.<br>5. Understand the principles of storage structure and recovery management. |
| **Assessment Methods:** | Skills will be assessed using a variety of strategies, including:<br>1.  Tests (open-answer or multiple choice) or quizzes to assess "knowing" and "remembering" skills.<br>2.  Group/individual projects, portfolios, case study, role playing<br>3.  Self- and peer-assessment, feedback (focus is on comments rather than grades)<br>4.  Labworks. An electronic report is prepared for each laboratory work. Each laboratory work is then evaluated with a maximum of 10 points. |
| **Teacher(s):** | Evdoshenko  O.I. |
| **Prerequisite subject(s):** | None |
| **Compulsory Literature:** | 1. Alekseev V.A. Fundamentals of design and implementation of databases [Electronic resource]: guidelines for conducting laboratory work on the course "Databases" / Alekseev V.A. - Electron. text data. - Lipetsk: Lipetsk State Technical University, EBS ASV, 2014. - 26 p.— Mode of access: http://www.iprbookshop.ru/55122.html.- EBS "IPRbooks"<br><br>2. Databases. Theory and practice of application [Electronic resource]: textbook/ A.L. Bogdanova [et al.]. - Electron. text data.— Khimki: Russian International Academy of Tourism, 2010. - 125 p. — Mode of access: http://www.iprbookshop.ru/14277.html.- EBS "IPRbooks"<br><br>3. Korolev V.T. Technology of database management [Electronic resource]: textbook/ Korolev V.T., Gontarev E.A., Black A.M. — Electron. text data.- Moscow: Russian State University of Justice, 2015. - 108 p. — Mode of access: http://www.iprbookshop.ru/45233.html.- EBS "IPRbooks"<br><br>4. Moldovanova O.V. Information systems and databases [Electronic resource]: textbook/ Moldovanova O.V. - Electron. text data. - Novosibirsk: Siberian State University of Telecommunications and Computer Science, 2014. - 178 p. — Mode of access: http://www.iprbookshop.ru/45470.html.- EBS "IPRbooks"<br><br>5. Medvedkova I.E. Databases [Electronic resource]: textbook/Medvedkov, I.E., Bugaev Y.V., Chikunov, S.V., Electron. text data. — Voronezh: Voronezh State University of Engineering Technologies, 2014. - 104 p. — Mode of access: http://www.iprbookshop.ru/47418.html.- EBS "IPRbooks"<br><br>6. Kondrashov, Yu.N. the SQL Language: a Collection of situational problems in the discipline "Databases: tutorial / Kondrashov Yu.N — Moscow: Rusains, 2018. - 125 p — ISBN 978-5-4365-2335-4. – URL: https://book.ru/book/929715 (accessed: 08.10.2019). - Text: electronic. |
| **Replacement Literature:** | 1. Literature on SQL server PostgreSQL [Electronic resource] - On-line library xServer.ru. Mode of access: http://www.xserver.ru/computer/database/postgresql/<br>2. Metanit.com - site on programming [Electronic resource] - access Mode: |

| | http://www.metanit.com |
|---|---|
| | 3. KNOW-Intuit / Courses in programming [Electronic resource] - access Mode: http://www.intuit.ru |
| | 4. Osipov, D.L. Database Management Systems: Workshop / Osipov, D.L., Ogur, M.G., comp. - Stavropol: North Caucasus Federal University, 2017. - 148 p. - URL: https://book.ru/book/929919 (accessed: 08.10.2019). - Text: electronic. |
| |       1.    5. Korolev, V.T. Technology of Database Management: Textbook / Korolev V.T., Kontarev E.A., Chernykh A.M. - Moscow: Russian State University of Justice, 2015. - 108 p — - ISBN 978-5-93916-470-2. - URL: https://book.ru/book/930867 (accessed: 08.10.2019). - Text: electronic. |
| **Participation and Exam requirements:** | Attendance is important. During lectures students should actively participate in discussions on the topic, and during laboratory classes – independently perform all the tasks presented. The completion of all laboratory work is the condition for sitting exam. The exam consists of two stages: answering questions and completing a practical assignment. |
| **Independent work:** | Independent work includes the implementation of an individual project within the course. The individual project consists of 4 stages: domain analysis, database development, development of the main interface forms of the application, and development of output documents. |
| **Date 1** | Lecture: "Introduction to database theory" |
| **Date 2** | Lecture: "Data Models. Relational model"<br>Lab work: "Creating a database in the PostgreSQL database management system" |
| **Date 3** | Lecture: "Operations of Relational Algebra"<br>Lab work: "Data Sampling using SQL queries in pgAdmin" |
| **Date 4** | Lecture: "Normalization of Relations in DB"<br>Lab work: "Using aggregate functions in SQL-queries" |
| **Date 5** | Lecture: "Introduction to MS SQL Server"<br>Lab work: "Grouping data using SQL-queries" |
| **Date 6** | Lecture: "Basics of T-SQL. DDL"<br>Lab work: "Creating views" |
| **Date 7** | Lab work: "Development of subqueries" |
| **Date 8** | Lecture: "Built-in functions. Variables and control constructs"<br>Lab work: "Creating stored procedures" |

WEB Programming

| **Subject code:** | **Subject name:** WEB Programming | | |
|---|---|---|---|
| **Study load:**<br>5 ECTS | **Load of contact hours:** | **Study semester:** | **Assessment:** |
| **Objectives:** | The goal of the course is to master basic knowledge, concepts and skills in the field of WEB application development. | | |
| **Course outline:** | The goal of the course can be achieved by combining a set of teaching methods, including lectures and laboratory work performed on a computer.<br><br>The lecture-based training is conducted in a traditional form. Computer presentation is required as a visual aid for the lecture classes.<br><br>Laboratory works on a computer are aimed to build practical skills. In the course of the work, the following goals are achieved: | | |

| | |
|---|---|
| | • students consolidate the theoretical knowledge gained during lectures and update its practical significance. Students feel more motivated to master the course;<br><br>• students acquire practical programming skills in high-level languages;<br><br>• students acquire basic skills in using software development environments;<br><br>• Students learn to identify and solve errors and emergency situations.<br><br>Topics covered:<br><br>1. Basic principles of web technologies implementation<br><br>2. HTML markup language.<br><br>3. The basics of CSS.<br><br>4. The PHP programming language<br><br>5. The Basics of MySQL<br><br>6. The JavaScript programming language<br><br>7. The use of CGI technology for web programming.<br><br>Additional training materials are available at online learning platform. Most classes are dedicated to teamwork and design workshops. |
| **Learning Outcomes:** | On completion of the course, students are expected to be able to:<br><br>1. Understand the concepts, principles and methods of Web Programming.<br>2. Apply the concepts, principles, and methods of Web Programming to Web applications development.<br>3. Be familiar with current Web technologies.<br>4. Understand the technologies, business models and societal issues of Web 2.0 |
| **Assessment Methods:** | Skills will be assessed using a variety of strategies, including:<br>1. Tests (open-answer or multiple choice) or quizzes to assess "knowing" and "remembering" skills.<br>2. Group/individual projects, portfolios, case study, role playing<br>3. Self- and peer-assessment, feedback (focus is on comments rather than grades)<br>4. Labworks<br>The final grade is given after passing the final test. |
| **Teacher(s):** | Morozov B.B. |
| **Prerequisite subject(s):** | None |
| **Compulsory Literature:** | Sychev A.V., Web-technologies / Sychev A.V.-M.: National Open University "INTUIT", 2016. - Text: electronic / / Electronic Library System "Student Consultant": [site]. - URL : http://www.studentlibrary.ru/book/intuit018.html (accessed: 14.02.2020). - Mode of access : by subscription.<br><br>Chumikov A., PR technologies on the Internet: Web 1.0, Web 2.0, Web 3.0 / Chumikov A., Bocharov M., Tishkova M.-M.: Alpina publisher, 2016. - ISBN 978-5-9614-2051-7-Text: electronic / / Electronic Library System "Student Consultant": |

| | |
|---|---|
| | [site]. - URL : http://www.studentlibrary.ru/book/ISBN9785961420517.html (accessed: 14.02.2020). - Mode of access : by subscription. |
| **Replacement Literature:** | Martinez A., Secrets of creating an inexpensive Web site. How to create and maintain a successful Web site without spending a penny / Martinez A.-M.: DMK Press, 2016. - 416 p. (Web design series) - ISBN 5-94074-162-2-Text: electronic / / Electronic Library System "Student Consultant": [website]. - URL : http://www.studentlibrary.ru/book/ISBN5940741622.html (accessed: 14.02.2020). - Mode of access : by subscription. |
| **Participation and Exam requirements:** | It is important to attend and actively participate in classes, as most classes will be devoted to laboratory works. |
| **Independent work:** | Independent work includes the implementation of an individual project within the discipline. |
| **1) Date 1** | Laboratory work 1 Design an information system based on the "File server" architecture for the management information system. |
| **2) Date 2** | Laboratory work 2 Writing a script "Tip of the Day". |
| **3) Date 3** | Laboratory work 3 Writing a script of a link checker that tracks the number of clicks on download links |
| **4) Date 4** | Laboratory work 4 Creating an asynchronous data exchange between the client and server parts using AJAX technology for web site |
| **5) Date 5** | Presentation of an individual project. |

Big Data Analytics

| **Subject code:** | **Subject name:** Big Data Analytics | | |
|---|---|---|---|
| **Study load:** 5 ECTS | **Load of contact hours:** | **Study semester:** | **Assessment:** |
| **Objectives:** | The goal of the course is to provide an overall understanding of the technologies for the preparation, storage, processing and analysis of big data, to form skills to work with large amounts of data. | | |
| **Course outline:** | The course focuses on techniques of working with big data. Topics covered include hands-on analysis of large amounts of information using machine learning methods and big data frameworks. <br><br> Topics covered: <br> 1. Big Data Ecosystem <br> 2. Data mining process <br> 3. Work with large datasets <br> 4. Hadoop and Spark <br> 5. Large-scale machine learning at Spark <br> 6. NoSQL <br> 7. Big data visualization <br><br> Special attention is paid to the infrastructural aspects of working with big data. Additional materials will be provided through the online platform. | | |
| **Learning Outcomes:** | On completion of the course, students are expected to be able to: | | |

|  |  |
|---|---|
|  | 1. Analyze legal and ethical principles applied to contexts and environments of data science and decision making.<br>2. Design tested and effective advanced analytics models and simulations for decision making.<br>3. Demonstrate knowledge of statistical data analysis techniques utilized in decision making.<br>4. Use data mining software to solve real-world problems.<br>5. Employ cutting edge tools and technologies to analyze Big Data. |
| **Assessment Methods:** | During the semester, students perform laboratory work, grades on which form a rating based on the results of the work. Labs include setting up an environment for work, learning algorithms, and presenting results. Students are to prepare an electronic report for each laboratory work. |
| **Teacher(s):** | Koshkarov A.V. |
| **Prerequisite subject(s):** | None |
| **Compulsory Literature:** | Leskovec J., Rajaraman A., Ullman J. D. Mining of massive data sets. – Cambridge university press, 2020. |
| **Replacement Literature:** | Karau H. et al. Learning spark: lightning-fast big data analysis. – " O'Reilly Media, Inc.", 2015. |
| **Participation and Exam requirements:** | Attending classes is an important aspect of training, as part of the work involves the exchange of experience and ideas on the implementation of projects (laboratory work). Labs also provide a deeper understanding of big data frameworks and Python libraries.<br>The final grade will be based on the results of laboratory work. |
| **Independent work:** | Additional work includes a deeper study of large-scale machine learning algorithms. This will help students learn algorithms and analyze data more efficiently. |
| **Date 1** | Lecture "Big Data Ecosystem" |
| **Date 2** | Lecture "Data Mining Process" |
| **Date 3** | Lab "Data Analytics Based on CRISP-DM" |
| **Date 4** | Lab "Installing and Configuring a Big Data System" |
| **Date 5** | Lecture "Introduction to Apache Hadoop" |
| **Date 6** | Lab "Working in Hadoop» |
| **Date 7** | Lecture "Introduction to Apache Spark" |
| **Date 8** | Lab "Spark Platform" |
| **Date 9** | Lab "Machine Learning at Spark" |
| **Date 10** | Lecture "NoSQL" |
| **Date 11** | Lab «NoSQL» |
| **Date 12** | Lab "Big Data Visualization" |

Software engineering

| **Subject code:** | **Subject name:** Software engineering |
|---|---|

| **Study load:** 4 ECTS | **Load of contact hours:** | **Study semester:** | **Assessment:** |
|---|---|---|---|
| **Objectives:** | The course is aimed at students' gaining of knowledge and skills in design, testing, adjustment, implementation and maintenance of information systems. | | |
| **Course outline:** | The course allows for the students to understand the life cycle of information systems, tasks that arise at different stages of the life cycle, to master the methods, means and technologies that are applied for systems analysis and information system development, to learn to develop and improve software for information systems within different models of the life cycle.<br><br>Topics covered:<br>1. Introduction to systems engineering<br>2. Systems approach and systems thinking<br>3. System life cycle<br>4. Systems engineering practice<br>5. Requirements engineering<br>6. Architecture design<br>7. Data-centric integration of data | | |
| **Learning Outcomes:** | On completion of the course, students are expected to be able to:<br><br>1. Identify, formulate, and solve software engineering problems, including the specification, design, implementation, and testing of software systems that meet specification, performance, maintenance and quality requirements;<br>2. Elicit, analyze and specify software requirements through a productive working relationship with various stakeholders of a software development project;<br>3. Understand professional, ethical and social responsibility of a software engineer;<br>4. Evaluate the impact of potential solutions to software engineering problems in a global society, using the knowledge of contemporary issues and emerging software engineering trends, models, tools, and techniques. | | |
| **Assessment Methods:** | Skills will be assessed using a variety of strategies, including:<br>1.      Tests (open-answer or multiple choice) or quizzes to assess "knowing" and "remembering" skills.<br>2.      Group/individual projects, portfolios, case study, role playing<br>3.      Self- and peer-assessment, feedback (focus is on comments rather than grades)<br>4.      Labworks. Students are to prepare an electronic report for each laboratory work. Each laboratory work is assessed on a 10-point scale. | | |
| **Teacher(s):** | Evdoshenko O.I. | | |
| **Prerequisite subject(s):** | None | | |
| **Compulsory Literature:** | 1   Software engineering. Theory and practice [electronic resource] / Antamoshkin, O.A. – Krasnoyarsk, Siberian Federal University, 2012. - http://www.studentlibrary.ru/book/ISBN9785763825114.html<br>2   Engineering of knowledge [electronic resource]: study guide / Zagorulko, Yu.A. – Novosibirsk: Publishing Centre of Novosibirsk State University, 2016. - http://www.studentlibrary.ru/book/ISBN9785443704524.html | | |

| | |
|---|---|
| | 3 Systems and software engineering. Glossary of terms [electronic resource]: study guide for universities / Batovrin, V.K. – Moscow: DMK Press, 2010. - http://www.studentlibrary.ru/book/ISBN9785940745921.html<br>4 Introduction to software engineering [electronic resource]: study guide / Soloviev, N.A. – Orenburg: Orenburg State University, 2017. - http://www.studentlibrary.ru/book/ISBN9785741016855.html<br>5 Batovrin, V.K. Systems and software engineering. Glossary of terms [electronic resource]: study guide for universities / Batovrin, V.K. – Electronic text data. – Saratov: ProfEducation, 2017.— 280 p.— Access mode: http://www.iprbookshop.ru/63956.html.— IPRbooks e-library |
| **Replacement Literature:** | 1. Engineering education design in the prospects of the 21st century [electronic resource]: study guide / A.I. Chuchalin – Moscow: Logos, 2017. - http://www.studentlibrary.ru/book/ISBN9785987047873.html<br>2. Productive practices of competence-based approach in education [electronic resource] monograph / executive secretary S.I. Osipova – Krasnoyarsk: Siberian Federal University, 2017. - http://www.studentlibrary.ru/book/ISBN9785763836363.html<br>3. Date, Christopher J. An Introduction to Database Systems. 7th edition. – Moscow: Williams Publishing House, 2002. - 1072 p. - ISBN 5-8459-0138-3: 308-52: 308-52.АБ-2;<br>4. Batovrin, V.K. Systems and software engineering. Glossary of terms [electronic resource]: study guide for universities / Batovrin, V.K. – Electronic text data. – Saratov: ProfEducation, 2017.— 280 p.— Access mode: http://www.iprbookshop.ru/63956.html.— IPRbooks e-library<br><br>5. Lipaev, V.V. Software engineering of complex custom software products [electronic resource]: study guide / Lipaev, V.V. – Electronic text data. – Moscow: MAKS Press, 2014.— 309 p.— Access mode: http://www.iprbookshop.ru/27297.html.— IPRbooks e-library |
| **Participation and Exam requirements:** | Attending classes is important. During the lectures, the students must participate actively in discussion of the issue; during the laboratory work, they complete all the given tasks independently. The students are allowed to take the examination only if they have completed all the laboratory works. The examination is held in two stages – answering the questions and completing a practical assignment. |
| **Independent work:** | The students prepare reports and summaries, made in electronic and printed forms, and it should contain objectives, necessary theoretical information in brief, and the obtained results and findings for each objective.<br><br>Examples of topics for reports and summaries, to be prepared by students in course of research under the guidance of the teacher.<br><br>• Case Studies (Guide to the Systems Engineering Body of Knowledge):<br><br>• Hubble Space Telescope Case Study<br><br>• Global Positioning System Case Study<br><br>• Medical Radiation Case Study<br><br>• FBI Virtual Case File System Case Study<br><br>• MSTI Case Study<br><br>• Next Generation Medical Infusion Pump Case Study. |

| | |
|---|---|
| **Date 1** | Lecture "Introduction to Systems Engineering"<br>Laboratory work "Calculation of the characteristics of a modular software system" |
| **Date 2** | Lecture "Systems approach and systems thinking"<br>Lab "Calculation of software system metrics" |
| **Date 3** | Lecture "System Life Cycle"<br>Lab "Software Testing - 1" |
| **Date 4** | Lecture "System Engineering Practice"<br>Lab "Software Testing - 2" |
| **Date 5** | Lecture "Requirements Engineering"<br>Laboratory work "Scrum-team formation" |
| **Date 6** | Lecture "Architectural Design"<br>Lab "Version Control System and Unit Tests" |
| **Date 7** | Lecture "Data-Centric Data Integration"<br>Laboratory work "Assembly and retrospective analysis" |

UI/UX design

| **Subject code:** | **Subject name:** UI/UX design | | |
|---|---|---|---|
| **Study load:**<br>4 ECTS | **Load of contact hours:** | **Study semester:** | **Assessment:** |
| **Objectives:** | The goal of the course is to develop students' practical skills in design and development of the user interface structure in mobile applications, taking into account consumer opinion and ergonomics requirements. | | |
| **Course outline:** | The course allows to familiarize the students with the main control elements of the mobile application graphical user interface, their properties, layout components, the requirements of interface ergonomics and styling of control elements of graphical user interface with these requirements; to acquire practical skills in designing graphical user interfaces for mobile applications, in cross-platform development of mobile applications on the Xamarin platform using various control elements and styles.<br><br>Topics covered:<br>1. Xamarin and cross-platform development<br>2. Graphical user interface in Xamarin Forms<br>3. Xamarin elements and their properties<br>4. Development of a database for data storage.<br>5. Resources and styles. Data binding in Xamarin<br>6. ListView and data management. Navigation<br>7. The types of pages. MVVM. FlexLayout | | |
| **Learning Outcomes:** | On completion of the course, students are expected to be able to:<br><br>1. Design user interfaces with target consumer in mind<br>2. Enumerate the important principles that underlie all good user interface design | | |

| | |
|---|---|
| | 3. Perform usability testing on newly designed interfaces |
| **Assessment Methods:** | Skills will be assessed using a variety of strategies, including:<br>1.     Tests (open-answer or multiple choice) or quizzes to assess "knowing" and "remembering" skills.<br>2.     Group/individual projects, portfolios, case study, role playing<br>3.     Self- and peer-assessment, feedback (focus is on comments rather than grades)<br>4.     Labworks. Students are to prepare an electronic report for each laboratory work. Each laboratory work is assessed on a 10-point scale. |
| **Teacher(s):** | Evdoshenko O.I |
| **Prerequisite subject(s):** | None |
| **Compulsory Literature:** | 1. Sokolova, V. V. Mobile Applications Development [electronic resource]: study guide / Sokolova, V. V. — Electronic text data. — Tomsk: Tomsk Polytechnic University, 2014. — 176 p. — Access mode: http://www.iprbookshop.ru/34706.html.- IPRbooks e-library<br><br>2. Programming Manual for Xamarin Forms [electronic resource] - Access mode: https://metanit.com/sharp/xamarin/<br><br>3. Fedotenko, M.A. Mobile Applications Development. First Steps [Electronic resource] / Fedotenko, M.A. — Electronic text data. — Moscow:  BKL Publishers, 2019. — 336 p. — Access mode: http://www.iprbookshop.ru/89067.html. — IPRbooks e-library |
| **Replacement Literature:** | 1. NOU-Intuit / Programming Courses [electronic resource] — Access mode: http://www.intuit.ru<br><br>2. C# Programming, .NET Framework+ Platform [Electronic resource] — Access mode: http://professorweb.ru<br><br>3. Vereskun, D.M. Development of Mobile Applications for Business [electronic resource]: study guide / Vereskun, D.M. — Electronic text data.- Saratov: Yuri Gagarin State Technical University of Saratov, EBS ASV, 2012. — 51 p. — Access mode: http://www.iprbookshop.ru/76508.html. — IPRbooks e-library |
| **Participation and Exam requirements:** | Attending classes is important. During the lectures, the students must participate actively in discussion of the issue; during the laboratory work, they complete all the given tasks independently. The students are allowed to take the examination only if they have completed all the laboratory works. The examination is held in two stages – answering the questions and completing a practical assignment. |
| **Independent work:** | Students are to prepare an independent electronic report for each laboratory work. Students prepare presentations on different topics related to the course. |
| **Date 1** | Lecture "Xamarin and cross-platform development"<br>Laboratory work "Mobile application development" |
| **Date 2** | Lecture "Graphical user interface in Xamarin Forms" |

| | |
|---|---|
| | Laboratory work "Development of navigation between pages" |
| Date 3 | Lecture "Xamarin elements and their properties" |
| | Laboratory work "Pop-up window development" |
| Date 4 | Lecture "Development of a database for data storage" |
| | Laboratory work "Development of a database for data storage" |
| Date 5 | Lecture "Resources and styles. Data binding in Xamarin" |
| | Laboratory work "Development of mobile applications styling (with CSS application)" |
| Date 6 | Lecture "ListView and data management. Navigation" |
| | Laboratory work "Trigger development" |
| Date 6 | Laboratory work "Development of visual component for search" |
| Date 7 | Lecture "The types of pages. MVVM. FlexLayout" |
| | Laboratory work "Development of a tabbed page" |

| Subject code: | Subject name: Economics and project management | | | |
|---|---|---|---|---|
| Study load:<br>4 ECTS | Load of contact hours: | Study semester: | Assessment: | |
| Objectives: | The goal of the course is to give a systematic understanding of how such projects work in financial, economic, social and environmental terms, to form students ' system of theoretical knowledge and practical skills in the formulation of economic and mathematical models, their analysis and use for making managerial decisions. | | | |
| Course outline: | The course allows to familiarize the students with mathematical methods and approaches used in the management and organization of business projects; to acquire practical skills in using economic and mathematical methods in business projects.<br><br>Topics covered:<br>Economic and mathematical model and its main elements.<br>Stages of modeling and development of management decisions.<br>Market assessment of the project.<br>Modeling and forecasting of market trends.<br>Collection and analysis of statistical data.<br>Visualization and interpretation of results.<br>Market potential assessment.<br>Methods of competitive analysis.<br>Overview of project monetization models.<br>Collecting and analyzing requirements.<br>Selecting and calculating key project metrics.<br>Fundamentals of financial modeling.<br>Model the flow of income and expenses.<br>Investment management models.<br>Business modeling and business model map. | | | |
| Learning Outcomes: | On completion of the course, students are expected to be able to:<br>1. Demonstrate understanding of the range of economic and management tools used in civil and environmental engineering projects and the research and **advanced scholarship** that produced these tools. | | | |

| | |
|---|---|
| | 2. Critically compare alternative approaches to engineering project economics and management. |
| | 3. Communicate the outcomes of financial, economic and managerial analyses to project clients. |
| | 4. Compare various decision-making tools that can be used in project analysis. |
| **Assessment Methods:** | Skills will be assessed using a variety of strategies, including:<br><br>1. Tests (open-answer or multiple choice) or quizzes to assess "knowing" and "remembering" skills.<br>2. writing discussion board postings or essays to assess "understanding" skills.<br>3. students will be given solutions to related problems and be asked to adapt them to new problems to assess "applying" and "analyzing" skills. (problem-solving exercises)<br>4. students will be asked to create and implement new solutions to assess their "creating" skills.<br>5. Group/individual projects, portfolios, case study, role playing<br>6. Self- and peer-assessment, feedback (focus is on comments rather than grades) |
| **Teacher(s):** | |
| **Prerequisite subject(s):** | None |
| **Compulsory Literature:** | Morris, P.W.G (2013). Reconstructing Project Management. |
| **Replacement Literature:** | Smith, N.J. (2008). Engineering Project Management. |
| **Participation and Exam requirements:** | Attending classes is important. During the lectures and presentations, the students must participate actively in discussion of the topic; during the seminars and doing practical assignments, they complete all the given tasks independently or in mini groups (pairs). The students are allowed to take the final test only if they have completed all the previous tasks. The final test consists of two parts – answering theoretical questions and completing a practical assignment based on case study. |
| **Independent work:** | Students prepare presentations on different topics related to the course. Independent work also includes the implementation of an individual project within the discipline. |
| **Date 1** | 1. The Project and Policy Life Cycles. |
| **Date 2** | 2. Financial Decision Making and Discounting. |
| **Date 3** | 3. Present Worth and Equivalent Annual Worth. |
| **Date 4** | 4. Rate of Return and Benefit Cost Ratios. |
| **Date 5** | 5. Cost Benefit Analysis: Theory and Practice. |
| **Date 6** | 6. Replacement Decisions. |
| **Date 7** | 7. Project Financing and Capital Budgeting. |
| **Date 8** | 8. Model the flow of income and expenses. Investment management models. Business modeling and business model map. |

| Date 9 | 9. Multi Criteria Analysis and Related Methods. |
|---|---|
| Date 10 | 10. Approaches to Risk and Uncertainty. |
| Date 11 | 11. Public Private Partnerships. |
| Date 12 | 12. Project Costs. |

| Subject code: | Subject name: Psychology of personal and team efficiency | | |
|---|---|---|---|
| Study load:<br>4 ECTS | Load of contact hours: | Study semester: | Assessment: |
| Objectives: | The goal of the course is to improve so-called soft skills of developing complex IT projects for various purposes. | | |
| Course outline: | The course allows to form skills of independent search for missing information; to form an ability to use the acquired knowledge to solve cognitive and practical problems; to acquire communication skills when working in groups; to develop research skills (identifying problems, collecting information, observing, conducting experiments, analyzing, hypothesizing, generalizing) and the system of critical thinking in general.<br><br>Topics covered:<br>Analysis of the subject area.<br>Development of an investment IT project in a given subject area.<br>Modeling in a given subject area.<br>A comprehensive software IT project based on the integration of various technologies in a given subject area. | | |
| Learning Outcomes: | On completion of the course, students are expected to be able to:<br>1. Recognise the five stages of team development<br>2. Inspire and motivate team members<br>3. Build teamwork by applying the twelve characteristics of an effective team<br>4. Develop strategies for dealing with team conflict and common problems<br>5. Communicate effectively<br>6. Engage in active listening<br>7. Build consensus through understanding | | |
| Assessment Methods: | Skills will be assessed using a variety of strategies, including:<br><br>1. Tests (open-answer or multiple choice) or quizzes to assess "knowing" and "remembering" skills.<br>2. writing discussion board postings or essays to assess "understanding" skills.<br>3. students will be given solutions to related problems and be asked to adapt them to new problems to assess "applying" and "analyzing" skills. (problem-solving exercises)<br>4. students will be asked to create and implement new solutions to assess their "creating" skills.<br>5. Group/individual projects, portfolios, case study, role playing<br>6. Self- and peer-assessment, feedback (focus is on comments rather than | | |

| | grades) |
|---|---|
| **Teacher(s):** | |
| **Prerequisite subject(s):** | None |
| **Compulsory Literature:** | Levi, Daniel (2014). Group Dynamics for Teams (5th Edition). Los Angeles, California: Sage Publications. ISBN: 978-1-4833-7834-3 |
| **Replacement Literature:** | James Cain and Barry Jolliff (2010). Teamwork and Teamplay (1st Edition) . Kendall Hunt Publishing. ISBN-13: 978-0787245320 |
| **Participation and Exam requirements:** | Attending classes is important. During the lectures and presentations, the students must participate actively in discussion of the topic; during the seminars and doing practical assignments, they complete all the given tasks independently or in mini groups (pairs). The students are allowed to take the final test only if they have completed all the previous tasks. The final test consists of two parts – answering theoretical questions and completing a practical assignment based on case study. |
| **Independent work:** | Students prepare presentations on different topics related to the course. Independent work also includes the implementation of an individual project within the discipline. |
| **Date 1** | Teams and team working |
| **Date 2** | Decision-making and managing team-based conflicts |
| **Date 3** | Relationship building |
| **Date 4** | Fostering critical self-reflection |
| **Date 5** | Managing performance |
| **Date 6** | Leading team strategies into action |
| **Date 6** | Process of Teamwork |
| **Date 7** | Cooperation & Competition |

| **Subject code:** | **Subject name:** Specifics of Android Development | | |
|---|---|---|---|
| **Study load:** 5 ECTS | **Load of contact hours:** | **Study semester:** | **Assessment:** |
| **Objectives:** | The purpose of the course is to form students' knowledge and practical skills in the field of mobile application development using programming technologies based on the Android platform. | | |
| **Course outline:** | The course focuses on developing applications for mobile devices based on Android. Special attention is paid to the IDE Android Studio programming environment. Topics covered: 1. Creating a simple application 2. Creating a simple interactive application 3. Using widgets 4. Using multiple activity. Work with sound. 5. Lists. Custom adapters. 6. Work with network resources. | | |

| | Special attention is paid to the IDE Android Studio programming environment. |
|---|---|
| **Learning Outcomes:** | On completion of the course, students are expected to be able to:<br><br>1. Design and create Android apps.<br>2. Use the Java programming language, the Android SDK, and Android Studio developer tools.<br>3. Gain fundamental knowledge essential to not only Android development, but mobile development in general. |
| **Assessment Methods:** | Skills will be assessed using a variety of strategies, including:<br>1. Tests (open-answer or multiple choice) or quizzes to assess "knowing" and "remembering" skills.<br>2. Group/individual projects, portfolios, case study, role playing<br>3. Self- and peer-assessment, feedback (focus is on comments rather than grades)<br>4. Labworks. Students are to prepare an electronic report for each laboratory work. |
| **Teacher(s):** | Denis Zholobov |
| **Prerequisite subject(s):** | None |
| **Compulsory Literature:** | 1. Kamaev V. A. Programming technologies: a textbook for university students. - M.: Higher. school, 2006 .-- 454 p.<br>2. Horev P.B. Technologies of object-oriented programming. - M.: Academy, 2004 .-- 448 p. |
| **Replacement Literature:** | Griffiths D., Griffiths D. Head First Android Development: a brain-friendly guide. – " O'Reilly Media, Inc.", 2017. |
| **Participation and Exam requirements:** | Attending classes and completing assignments is mandatory. The final grade is set based on the results of laboratory work. |
| **Independent work:** | Additional work includes the use of additional electronic sources for programming in Java and Android development. Each laboratory work involves the search for additional information |
| **Date 1** | Lab "Creating a simple application" |
| **Date 2** | Lab "Creating a simple interactive application" |
| **Date 3** | Lab "Using Multiple Activities. Work with sound" |
| **Date 4** | Lab "Lists. Custom Adapters" |
| **Date 5** | Lab "Work with network resources" |

| **Subject code:** | **Subject name:** Neural networks models for mobile applications | | |
|---|---|---|---|
| **Study load:**<br>5 ECTS | **Load of contact hours:** | **Study semester:** | **Assessment:** |
| **Objectives:** | The goal of the course is to form students' knowledge and practical skills in the field of Neural networks models in mobile applications. | | |
| **Course outline:** | The course is offered as a combination of theoretical (lectures) and practical (seminars/workshops/labs) parts. Students design their own models of Neural | | |

|  | Network and improve them from simple to complex during the course. After the completion of each thematic section (chapter), they are to apply their new knowledge to the model and thus see the improvement in practice. Using several versions of each hardware platform for deploying one app enables students to resolve some particular problems by examining existing solutions on mobile NN society.<br>The course focuses on NN models for a lower-powered, battery-operated devices, on both Android and iOS platforms.<br><br>Topics covered:<br>   1. Usage of TensorBoard to evaluate models.<br>   2. Retraining deployed models with user data while maintaining data privacy. Deploying NN on embedded systems using TensorFlow on Raspberry Pi and microcontrollers. |
| --- | --- |
| **Learning Outcomes:** | On completion of the course, students are expected to be able to:<br><br>   1. Optimize neural networks for mobile use, with respect to the low battery power and processing power.<br>   2. Build applications on Android and iOS that use the neural networks by using the TensorFlow Lite Interpreter in these environments.<br>   3. Understand embedded systems and microcontrollers, running the models on Raspberry Pi and SparkFun Edge boards. |
| **Assessment Methods:** | Skills will be assessed using a variety of strategies, including:<br>1.     Tests (open-answer or multiple choice) or quizzes to assess "knowing" and "remembering" skills.<br>2.     Group/individual projects, portfolios, case study, role playing<br>3.     Self- and peer-assessment, feedback (focus is on comments rather than grades)<br>4.     Labworks |
| **Teacher(s):** | Veriga A.V. |
| **Prerequisite subject(s):** | Machine Learning |
| **Compulsory Literature:** | Goodfellow, Y. Bengio, and A. Courville, Deep Learning. Cambridge, MA, USA: The MIT Press, 2016. |
| **Replacement Literature:** | Bishop, Ch., "Neural Networks for Pattern Recognition", 1996, Oxford University Press, USA; 1 edition, ISBN-13: 978-0198538646. |
| **Participation and Exam requirements:** | Attending classes and completing assignments is mandatory as most of the classes are dedicated to team works and workshops. Some experiments to deploy models on different platforms are most optimal on a few emulators simultaneously with different hyper-parameters and succeeding discussion. |
| **Independent work:** | Additional work includes the use of additional electronic sources for Neural networks models for mobile applications Each practical assignment involves the search for additional information on the following topics:<br>TensorFlow Lite GPU delegate for cross platform acceleration (additional topic)<br>The Coco dataset (self study)<br>Object detection model trained on the Coco (practices) |

| | |
|---|---|
| | Rock Paper Scissors for Android (practices)<br>Rock, Paper, Scissors on iOS (practices)<br>Rock Paper Scissors on Raspberry Pi (practices)<br>https://tensorboard.dev/(self exploring) |
| **1) Date 1** | Device-based models with TensorFlow Lite.<br>Features and components of mobile AI. Architecture and performance.<br>Optimization Techniques. |
| **2) Date 2** | Quantization. TF-Select. Converting a model to TFLite. Transfer learning with TFLite. |
| **3) Date 3** | Running a TF model in an Android app. Installation and resources. Architecture of a model. Initializing the Interpreter. Preparing the input. Inference and results. |
| **4) Date 4** | Classifying camera images. Initialize and prepare input. Code of camera image classifier. Initialize model and prepare inputs. Run the object detection app.Code for inference and results. |
| **5) Date 5** | Building the TensorFLow model on IOS. Tensorflow Lite Swift. Cats vs Dogs App. Taking the initial steps. Scaling the image. Looking at the App in Xcode |
| **6) Date 6** | The code for the image classification App. Object classification intro. Code for the object detection model |
| **7) Date 7** | TensorFlow Lite on devices. Starting to work on a Raspberry Pi. Image classification The 4 step process. Object detection. |
| **Date 8** | Raspberry Pi demo. Microcontrollers. Pre optimized mobileNet. Object detection model trained on the coco. |
| **Date 9** | Tensorboard: tools for model training. Tensorboard scalars. Callbacks. Histograms. Publishing model details. Local tensorboard. Looking at graphics in a dataset. More than one image. Confusion matrix. Multiple callbacks |
| **Date 10** | Federated Learning. Training on mobile devices. Data at the edge. Maintaining user privacy. Masking. APIs for Federated Learning. Example of federated learning. |

| | |
|---|---|
| **Subject code:** | **Subject name:** Mobile application security |

| **Study load:**<br>5 ECTS | **Load of contact hours:** | **Study semester:** | **Assessment:** |
|---|---|---|---|

| | |
|---|---|
| **Objectives:** | The purpose of the course is to get acquainted with existing security tools in Android applications. The course helps to acquire knowledge and form skills needed to develop Android applications based on information security requirements. |
| **Course outline:** | The course consists of a theoretical and practical part.<br>Within theoretical part, students learn the basic principles of developing secure mobile applications.<br>During practical classes, students learn to implement information security algorithms in mobile applications.<br>Topics covered:<br>    1. Android security architecture<br>    2. Android access rights differentiation<br>    3. Security components and data storage protection<br>    4. Client-server communication security |

| Learning Outcomes: | On completion of the course, students are expected to be able to:<br>1. Understand and communicate the mobile application security risks associated with hacking and other exploits<br>2. Discover real-world mobile application hacking techniques and countermeasures<br>3. Perform a mobile application penetration testing on own deployment and evaluate mobile application security threats and possible exploits<br>4. Develop and design a secure mobile application |
|---|---|
| Assessment Methods: | Skills will be assessed using a variety of strategies, including:<br><br>1. Tests (open-answer or multiple choice) or quizzes to assess "knowing" and "remembering" skills.<br>2. Group/individual projects, portfolios, case study, role playing<br>3. Labworks |
| Teacher(s): | Aleksei Podgornyi |
| Prerequisite subject(s): | None |
| Compulsory Literature: | 1. Mukherjea, Sougata. Mobile Application Development, Usability, and Security<br>2. Adetunji A. Olunuga. Profile Analysis of Mobile Application Security |
| Replacement Literature: | Gerardus Blokdyk. Mobile Application Security The Ultimate Step-By-Step Guide |
| Participation and Exam requirements: | Visiting the classes and active participation is important because most of the classes will be dedicated to team works and workshops and not for presentations. |
| Independent work: | Students prepare presentations on different topics related to the course.<br>Students search for additional information independently. |
| 1) Date 1 | Classroom presentation – Introduction to the course<br>Classroom presentation – Android application security architecture<br>Practice – Android app components |
| 2) Date 2 | Classroom presentation – Core security<br>Practice – Management and services |
| 3) Date 3 | Classroom presentation – Android Application Framework Layer<br>Practice – Using secure APIs |
| 4) Date 4 | Classroom presentation – Malware for Android: warning, detection, and elimination<br>Practice – Restricting access to components |
| 5) Date | Classroom presentation – Enhanced security<br>Practice – Encryption and decryption |
| 6) Date | Classroom presentation – Vulnerabilities in data warehouses<br>Practice – The application signature |
| 7) Date | Classroom presentation – Protection of data transmission over the network<br>Practice – Working with network in mobile applications |

| 8) **Date** | Classroom presentation – Preventing the forced introduction of malicious programs<br>Practice – Input validation |
|---|---|

| **Subject code:** | **Subject name:** AR and VR Technologies for Mobile Applications | | |
|---|---|---|---|
| **Study load:**<br>4 ECTS | **Load of contact hours:** | **Study semester:** | **Assessment:** |
| **Objectives:** | The goal of the course is to develop skills in developing mobile applications using virtual and augmented reality technologies. | | |
| **Course outline:** | The course focuses on an idea of the structure, characteristics, and varieties of systems used in AR and VR environments; it helps to master students' skills of analyzing various models used in AR and VR; skills of working with separate models of AR and VR systems.<br>Topics covered:<br>Basic concepts of AR and VR. AR and VR objects.<br>AR and VR systems.<br>AR and VR tools.<br>Application of AR and VR technologies in the development of mobile technologies. | | |
| **Learning Outcomes:** | On completion of the course, students are expected to be able to:<br>1. Demonstrate knowledge in VR and AR technologies in terms of used devices, building of the virtual environment and modalities of interaction and modelling.<br>2. Demonstrate knowledge in the main application of VR and AR technologies in medicine and surgery, cultural heritage, mobile applications and games. | | |
| **Assessment Methods:** | Skills will be assessed using a variety of strategies, including:<br><br>4. Tests (open-answer or multiple choice) or quizzes to assess "knowing" and "remembering" skills.<br>5. writing discussion board postings or essays to assess "understanding" skills.<br>6. students will be given solutions to related problems and be asked to adapt them to new problems to assess "applying" and "analyzing" skills. (problem-solving exercises)<br>7. students will be asked to create and implement new solutions to assess their "creating" skills.<br>8. Group/individual projects, portfolios, case study, role playing<br>9. Self- and peer-assessment, feedback (focus is on comments rather than grades) | | |
| **Teacher(s):** | Morozov B.B. | | |
| **Prerequisite subject(s):** | None | | |

| | |
|---|---|
| **Compulsory Literature:** | Virtual Reality Technology G. Burdea and P. Coiffet John Wiley & Sons, Inc. Emerging Technologies of Augmented Reality M. Haller, M. Billinghurst and B. Thomas Idea Group Publishing |
| **Replacement Literature:** | Introduction to Virtual Reality J. Vince Springer Augmented Reality R. Behriger, G. Klinker and D. W. Mizell A K Peters |
| **Participation and Exam requirements:** | Students are to prepare an individual project on the building of a simple virtual environment which will be presented during the oral exam. |
| **Independent work:** | Students prepare presentations on different topics related to the course. Students search for additional information independently. |
| **Date 1** | Introduction to Virtual Reality technology |
| **Date 2** | Introduction to Augmented Reality technology |
| **Date 3** | Visualization devices o Head Mounted Display o Cave o 3D display |
| **Date 4** | Building of the virtual environment o VR engine software: XVR |
| **Date 5** | Graphics rendering |
| **Date 6** | Interaction in the virtual environment o Collision detection o Collision response o Force feedback o Haptic interfaces |
| | Physical modelling |
| **Date 6** | Virtual Reality in medicine and surgery |
| **Date 7** | Virtual Reality in Cultural Heritage |


| **Subject code:** | **Subject name:** Cloud services for mobile development | | |
|---|---|---|---|
| **Study load:** 5 ECTS | **Load of contact hours:** | **Study semester:** | **Assessment:** |
| **Objectives:** | The purpose of the course is to provide students with knowledge and skills in the field of modern cloud technologies and services and their use in the gaming industry, to form an understanding of their capabilities when creating games, to master practical skills of working with cloud services. | | |
| **Course outline:** | The objectives of the course are achieved by combining a set of teaching methods, including lectures and laboratory works performed on a computer. | | |

Topics covered:

1. App publications in the cloud, overview of the Azure SDK.
2. Connecting a web role to a local network and managing the web role in PowerShell.
3. Standalone cloud application development;
4. Basic web service tools provided by the cloud computing concept.
5. Infrastructure - as-A-Service (IaaS).
6. Demonstration of creating a cloud solution project.
7. Features of its launch and state control using the Compute Emulator.

Additional training materials are distributed through the online training platform. Most classes are dedicated to teamwork and design workshops.

| | |
|---|---|
| **Learning Outcomes:** | On completion of the course, students are expected to be able to: <br><br> 1. Classify and describe the architecture and taxonomy of parallel and distributed computing, including shared and distributed memory, and data and task parallel computing. <br> 2. Characterize the distinctions between Infrastructure, Platform and Software as a Service (IaaS, PaaS, SaaS) abstractions, and Public and Private Clouds, and analyze their advantages and disadvantages. <br> 3. Compare synchronous and asynchronous execution patterns. <br> 4. Describe and compare different performance metrics for evaluating Cloud applications and demonstrate their use for application measurement. |
| **Assessment Methods:** | Skills will be assessed using a variety of strategies, including: <br> 1. Tests (open-answer or multiple choice) or quizzes to assess "knowing" and "remembering" skills. <br> 2. Group/individual projects, portfolios, case study, role playing <br> 3. Labworks. Interim Assessment of knowledge and skills obtained is carried out based on the results of laboratory work. <br><br> The final assessment is made after passing the final testing. |
| **Teacher(s):** | Morozov B.B. |
| **Prerequisite subject(s):** | None |
| **Compulsory Literature:** | Google Application Secrets / Denis Baluev. - M.: Alpina Publishers, 2010. - 287 p. http://www.studentlibrary.ru <br> Vladimir Safonov. Microsoft Windows Azure Cloud Computing Platform: Internet University of Information Technology, Binom. Knowledge Lab. ISBN 978-5-9556-0138-0, 978-5-9963-1223-8; 2013. - 240 p. <br> George Reese. Cloud Computing: BKhV-Peterburg. ISBN 978-5-9775-0630-4; 2011. 288 p. |
| **Replacement Literature:** | Mikheev M.O. VMware vSphere 4.1 Administration - M.: DMK Press, 2011. - 448 p. http://www.studentlibrary.ru |
| **Participation and Exam requirements:** | Visiting the classes and active participation is important because most of the classes will be dedicated for team works and labs. |
| **Independent work:** | A short project has to be developed from the students on the building of a simple virtual environment <br> A presentation of this project has to be presented during the oral exam. |
| **1) Date 1** | Laboratory work No. 1 Creating a remote server (Windows or Linux optional) |
| **2) Date 2** | Laboratory work No. 2 Creating an automatic web server load balancer. |
| **3) Date 3** | Laboratory work No. 3 Creating a web application using the EC2 AWS service and configure its horizontal scaling. |
| **4) Date 4** | Laboratory work No. 4 Creating a virtual network consisting of 6 subnets |

| | located in 2 accessible zones of your chosen region. |
|---|---|
| **5) Date 5** | Laboratory work No. 5 Creating storages to configure them so that data stored in one storage is automatically saved in the second |
| **6) Date 6** | Laboratory work No. 6 Creating an ML model based on your own tables. |
| **7) Date 7** | Laboratory work No. 7 Lambda function. Creating the "Bucket Manager" application |
| | |