

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ АВТОМОБІЛЬНО-ДОРОЖНИЙ
УНІВЕРСИТЕТ**

Кафедра комп'ютерних технологій і мехатроніки

ЗАТВЕРДЖЕНО

Гарант освітньо-професійної програми
«Інженерія програмного забезпечення»
першого(бакалаврського) рівня освіти:
зав. каф. КТМ, д.т.н., проф.


Ніконов О.Я.

**СИЛАБУС
АНАЛІЗ ВИМОГ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ/
ANALYSIS OF SOFTWARE REQUIREMENTS
SYLLABUS**

освітній ступінь	бакалавр / bachelor
галузь знань	12 Інформаційні технології / Information Technology
спеціальність	121 Інженерія програмного забезпечення / Software Engineering
освітня програма	Програмне забезпечення систем / Systems Software

Харків 2020

Автор: Шапошнікова Олена Павлівна, доцент кафедри комп'ютерних технологій і мехатроніки

Силабус розглянуто та затверджено на засіданні кафедри комп'ютерних технологій і мехатроніки, протокол № 18 від «27» червня 2020 р.

СИЛАБУС

АНАЛІЗ ВИМОГ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ/

ANALYSIS OF SOFTWARE REQUIREMENTS

SYLLABUS

освітній ступінь	бакалавр / bachelor
галузь знань	12 Інформаційні технології / Information Technology
спеціальність	121 Інженерія програмного забезпечення / Software Engineering
освітня програма	Програмне забезпечення систем / Systems Software

Анотація курсу

1. Викладачі

1.1. Лектор: Шапошнікова Олена Павлівна

- доцент кафедри комп'ютерних технологій та мехатроніки;
- педагогічний стаж – 25 років
- контактний телефон +38-057-707-37-43
- e-mail: shaposhnikovaer@gmail.com
- наукові інтереси: архітектура ПЗ, аналіз вимог до ПЗ, управління ІТ проектами, якість ПЗ та тестування.

1.2. Асистент лектора:

2. Дисципліна «Аналіз вимог до програмного забезпечення»

- рік навчання: 2;
- семестр навчання: 4;
- кількість годин за семестр: 150, в т.ч.
лекційних: 16;
практичних занять: 32;
на самостійне опрацювання: 72;
- кількість аудиторних годин на тиждень
лекційних: 2 (раз на два тижні);
практичних занять: 2.

3. Час та місце проведення

- аудиторні заняття – відповідно до розкладу ХНАДУ, ауд. 214, 216;
- позааудиторна робота – самостійна робота студента із використанням технологій віртуалізації MS Visio, Draw.io, Visual Use case.

4. Пререквізити та постреквізити навчальної дисципліни:

- **перереквізити:** «Основи інформаційних технологій», «Алгоритмізація та програмування», «Операційні системи», «Об'єктно-орієнтоване програмування», «Алгоритми і структури даних».
- **постреквізити:** «Архітектура ПЗ», «Програмування баз даних», «Web-програмування», «Управління ІТ проектами», «Якість програмного забезпечення та тестування».

5. Характеристика дисципліни:

5.1. Призначення навчальної дисципліни: вивчення дисципліни «Аналіз вимог до програмного забезпечення» займає вагомe місце у структурі отримання знань за освітньою програмою «Програмне забезпечення систем». Процес проектування програмного забезпечення (ПЗ) починається із вивчення предметної області, формулювання концепції проекту, визначення його границь, виявлення зацікавлених осіб та формулювання вимог до ПЗ. Для того, щоб правильно спроектувати програмний продукт, потрібно визначити, що він буде робити, тобто необхідно спроектувати функціональну модель, тому першим кроком для опису функціональності системи є моделювання вимог до неї. Дисципліна «Аналіз вимог до програмного забезпечення» формує системний підхід до проектування ПЗ та інженерний світогляд фахівця у галузі ІТ.

5.2. Мета вивчення дисципліни: «Аналіз вимог до програмного забезпечення» є теоретична та практична підготовка студентів, яка має забезпечити отримання ними основних знань у галузі сучасних технологій

проектування, інженерії вимог до програмного забезпечення (ПЗ), отримання практичних навичок реалізації програмних систем, основи моделювання і аналізу програмних систем, аналізу розробки, специфікації та управління вимогами.

5.3. Задачі вивчення дисципліни: є формування сукупності знань щодо етапів та фаз життєвого циклу програмного продукту; методів аналізу та синтезу інформаційних систем; моделей бізнес-процесів; об'єктно-орієнтований підхід; аналіз структур інформаційних систем; механізми інтеграції систем, методології моделювання структури та поведінки програмних систем застосовуванням уніфікованої мови моделювання UML для побудови діаграм, уміння ставити задачу та використовувати сучасні технології розробки ПЗ.

Професійні компетентності, які отримують студенти після вивчення навчальної дисципліни:

Інтегральна компетентність:

здатність використовувати поглиблені теоретичні та фундаментальні знання, уміння і навички для успішного розв'язування спеціалізованих та практичних задач під час професійної діяльності у галузі інформаційних технологій.

Загальні компетентності:

- здатність до пошуку, оброблення та аналізу інформації з різних джерел.

Спеціальні (фахові, предметні) компетентності:

- здатність формулювати та забезпечувати вимоги щодо якості програмного забезпечення у відповідності з вимогами замовника, технічним завданням та стандартами.
- здатність здійснювати процес інтеграції системи, застосовувати стандарти і процедури управління змінами для підтримки цілісності, загальної функціональності і надійності програмного забезпечення.

Програмні результати навчання:

- знати основні процеси, фази та ітерації життєвого циклу програмного забезпечення.
- знати та вміти використовувати методи та засоби збору, формулювання та аналізу вимог до програмного забезпечення.
- вибирати вихідні дані для проектування, керуючись формальними методами опису вимог та моделювання.
- вміти проводити розрахунок економічної ефективності програмних систем.

5.4. Зміст навчальної дисципліни: відповідає навчальній та робочій програмі, яка відповідає запитам роботодавців.

5.5. План вивчення дисципліни

Результати навчання	Навчальна діяльність	Робочий час студента (год.)	Оцінювання (бал)
1	2	3	4
РОЗДІЛ 1. АНАЛІЗ ВИМОГ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ			
Тема 1. Вступ до проблеми розробки вимог		15	11
<p><i>Загальні та спеціальні компетентності:</i> здатність формулювати та забезпечувати вимоги щодо якості програмного забезпечення відповідності вимогами замовника, технічним завданням та стандартами.</p> <p><i>Результати навчання:</i> знати основні процеси, фази та ітерації життєвого циклу програмного забезпечення; вибирати вихідні дані для проектування, керуючись формальними методами опису вимог та моделювання. вміти проводити розрахунок економічної ефективності програмних систем.</p>	<p>Лекція №1. Вступ до проблеми розробки вимог <i>План лекції:</i> 1. Контекст управління вимогами до ПЗ. 2. Роль аналітика. 3. Вимоги. 4. Процес розробки вимог. Список рекомендованих джерел: Основний: 1-5 Додатковий: 1-3 Інтернет-ресурси: 1-3</p>	2	
	<p>Завдання для самостійної роботи: Вивчення та доповнення матеріалів лекції на основі самостійного опрацювання основних літературних джерел, зазначених у списку на електронних матеріалів, які надаються викладачем</p> <p>Питання, внесені на самостійне опрацювання: Поняття про предметну область проекту. 1. Формальні визначення. 2. Класифікація моделей.</p>	9	
	<p>Практичне заняття № 1. Аналіз предметної області. <i>Мета:</i> Отримати навички виявлення реальних потреб замовника та з'ясування сенсу висловлених вимог, а також розроблення концепції проекту. <i>Завдання:</i> 1. Ознайомитися з видами моделей. 2. Ознайомитися із методологіями IDEF. 3. Вивчити синтаксис IDEF0-моделей. <i>План заняття:</i> –Актуалізація теоретичного матеріалу –Виконання завдань практичної роботи –Презентація виконаної роботи</p>	4	
Тема 2. Виявлення вимог проекту		15	13
<p><i>Загальні та спеціальні</i></p>	<p>Лекція 2. Виявлення вимог проекту <i>План лекції:</i></p>	2	

<p><i>компетентності:</i> здатність до пошуку, оброблення та аналізу інформації з різних джерел; здатність формулювати та забезпечувати вимоги щодо якості програмного забезпечення у відповідності з вимогами замовника, технічним завданням та стандартами.</p> <p><i>Результати навчання:</i> знати та вміти використовувати методи та засоби збору, формулювання та аналізу вимог до програмного забезпечення. вибрати вихідні дані для проектування, керуючись формальними методами опису вимог та моделювання.</p>	<p>1. Виявлення вимог та потреб. 2. Робота з користувачами. 3. Робота зацікавленими особами. Список рекомендованих джерел: Основний: 1-7, 9 Додатковий: 1-2 Інтернет-ресурси: 1-4, 6</p>		
	<p>Завдання для самостійної роботи: Вивчення та доповнення матеріалів лекції на основі самостійного опрацювання основних літературних джерел, зазначених у списку на електронних матеріалів, які надаються викладачем</p> <p>Питання, внесені на самостійне опрацювання: Методології розроблення вимог до ПЗ</p> <p>1. Поняття методології розроблення ПЗ, 2. Вплив методології розробки ПЗ на процес розроблення вимог.</p>	9	
	<p>Практичне заняття № 2. Визначення концепції проекту. <i>Мета:</i> навчитися створювати модель системи в стандарті IDEF0. <i>Завдання:</i></p> <p>1. На основі вихідної ідеї сформулювати цілі і задачі проекту. 2. Розробити вихідне бачення – концепцію проекту. 3. Провести аналіз затребуваності майбутнього продукту. 4. Провести попереднє оцінювання ризиків майбутнього проекту.</p> <p><i>План заняття:</i> –Актуалізація теоретичного матеріалу –Виконання завдань практичної роботи –Презентація виконаної роботи</p>	4	
Тема 3. Збір первинних вимог		15	13
<p><i>Загальні та спеціальні компетентності:</i> здатність до пошуку, оброблення та аналізу інформації з різних джерел здатність формулювати та забезпечувати</p>	<p>Лекція 3. Збір первинних вимог. <i>План лекції:</i> 1. . Техніки для виявлення вимог та потреб. 2. Класифікація вимог. Список рекомендованих джерел: Основний: 2, 4, 5 Додатковий: 2, 7 Інтернет-ресурси: 8</p>	2	
	<p>Завдання для самостійної роботи:</p>	9	

<p>вимоги щодо якості програмного забезпечення у відповідності з вимогами замовника, технічним завданням та стандартами.</p> <p><i>Результати навчання:</i> знати та вміти використовувати методи та засоби збору, формулювання та аналізу вимог до програмного забезпечення. вибирати вихідні дані для проектування, керуючись формальними методами опису вимог та моделювання.</p>	<p>Вивчення та доповнення матеріалів лекції на основі самостійного опрацювання основних літературних джерел, зазначених у списку на електронних матеріалів, які надаються викладачем</p> <p>Питання, внесені на самостійне опрацювання: Case-засоби для розроблення вимог до ПЗ.</p> <ol style="list-style-type: none"> 1. Поняття про CASE-засоби для розроблення вимог; 2. Необхідні та достатні функції CASE-засобу для розроблення вимог; 3. Виконати аналітичний огляд CASE-засобів; 		
	<p>Практичне заняття № 3. Побудова функціональної моделі проекту. <i>Мета:</i> навчитися виявляти та аналізувати основні функції та складові частини ПС. <i>Завдання:</i></p> <ol style="list-style-type: none"> 1. Визначити головну бізнес-функцію проекту, 2. Визначити точку зору на модель, 3. Побудувати модель ПС «як є» та модель «як буде». <p><i>План заняття:</i> –Актуалізація теоретичного матеріалу –Виконання завдань практичної роботи –Презентація виконаної роботи</p>	4	
Тема 4. Концепція і границі програмної системи		15	11
<p><i>Загальні та спеціальні компетентності:</i> здатність до пошуку, оброблення та аналізу інформації з різних джерел.</p> <p><i>Результати навчання:</i> знати основні процеси, фази та ітерації життєвого циклу програмного забезпечення.</p>	<p>Лекція 4. Концепція і границі програмної системи. <i>План лекції:</i></p> <ol style="list-style-type: none"> 1. Концепція системи. 2. Зацікавлені особи. 3. Границі системи/ продукту. 4. Модель сценаріїв використання. <p>Список рекомендованих джерел: Основний: 1, 6, 7, 12 Додатковий: 1-5, 7 Інтернет-ресурси: 8</p>	2	
<p>знати та вміти використовувати методи та засоби збору, формулювання та</p>	<p>Завдання для самостійної роботи: Вивчення та доповнення матеріалів лекції на основі самостійного опрацювання основних літературних джерел, зазначених у списку на електронних матеріалів, які надаються викладачем</p>	9	

<p>аналізу вимог до програмного забезпечення. вибирати вихідні дані для проектування, керуючись формальними методами опису вимог та моделювання.</p>	<p>Питання, внесені на самостійне опрацювання: Методологія опису потоків робіт (IDEF3) 1. Синтаксис і семантика моделей IDEF3, 2. Діаграми, 3. Декомпозиція дій.</p>		
	<p>Практичне заняття № 4. Декомпозиція функціональної моделі <i>Мета:</i> навчитися виконувати декомпозицію контекстної діаграми. <i>Завдання:</i> 1. Визначити кількість підфункцій діаграми першого рівня. 2. Визначити кількість рівнів декомпозиції. 3. Виконати декомпозицію контекстної діаграми. <i>План заняття:</i> –Актуалізація теоретичного матеріалу –Виконання завдань практичної роботи –Презентація виконаної роботи</p>	4	
<p>Тема 5. Специфікація вимог до ПЗ</p>		15	12
<p><i>Загальні та спеціальні компетентності:</i> здатність до пошуку, оброблення та аналізу інформації з різних джерел; здатність формулювати та забезпечувати вимоги щодо якості програмного забезпечення у відповідності з вимогами замовника, технічним завданням та стандартами.</p>	<p>Лекція 5. Специфікація вимог до ПЗ. Мови специфікацій. Критерії хороших вимог. <i>План лекції:</i> 1. Функції системи, 2. Вимоги до даних, 3. Вимоги до зовнішніх інтерфейсів, 4. Критерії якісних вимог. Атрибути якості. 5. Мови специфікацій. Список рекомендованих джерел: Основний: 1-5, 9, 13 Додатковий: 6, 9 Інтернет-ресурси: 9</p>	2	
<p><i>Результати навчання:</i> знати основні процеси, фази та ітерації життєвого циклу програмного забезпечення. знати та вміти використовувати</p>	<p>Завдання для самостійної роботи: Вивчення та доповнення матеріалів лекції на основі самостійного опрацювання основних літературних джерел, зазначених у списку на електронних матеріалів, які надаються викладачем Питання, внесені на самостійне опрацювання: Методологія опису потоків даних (IDEF3) 1. Засоби моделювання функціональних вимог, 2. Призначення діаграми потоків даних,</p>	9	

методи та засоби збору, формулювання та аналізу вимог до програмного забезпечення. вибрати вихідні дані для проектування, керуючись формальними методами опису вимог та моделювання. вміти проводити розрахунок економічної ефективності програмних систем.	3. Розроблення діаграми потоків даних IDEF3.		
	Практичне заняття № 5. Розроблення дерева функцій проекту. <i>Мета: навчитися будувати дерево функцій.</i> <i>Завдання:</i> 1. Описати кореневу вершину та основні групи функцій системи, 2. Деталізувати кожен групу функцій, 3. Побудувати дерево функцій ПС. <i>План заняття:</i> –Актуалізація теоретичного матеріалу –Виконання завдань практичної роботи –Презентація виконаної роботи	4	
Тема 6. Документування вимог з використанням сценаріїв використання.		15	14
Загальні та спеціальні компетентності: здатність до пошуку, оброблення та аналізу інформації з різних джерел; здатність формулювати та забезпечувати вимоги щодо якості програмного забезпечення у відповідності з вимогами замовника, технічним завданням та стандартами. <i>Результати навчання:</i> знати та вміти використовувати методи та засоби збору, формулювання та аналізу вимог до програмного забезпечення. вибрати вихідні дані для проектування, керуючись	Лекція 6. Документування вимог з використанням сценаріїв використання. Бізнес-правила. Невраховані вимоги. <i>План лекції:</i> 1. Глосарій, 2. Документ-концепція, 3. Специфікація вимог, 4. Управління вимогами. Список рекомендованих джерел: Основний: 1-3, 8-13 Додатковий: 6, 9, 10, Інтернет-ресурси: 1-5	2	
	Завдання для самостійної роботи: Вивчення та доповнення матеріалів лекції на основі самостійного опрацювання основних літературних джерел, зазначених у списку на електронних матеріалів, які надаються викладачем Питання, внесені на самостійне опрацювання: Мова моделювання UML 1. Призначення мови UML, 2. Процес проектування в UML, 3. Групи діаграм.	9	
	Практичне заняття № 6. Розроблення вимог до проекту: історії користувача <i>Мета: навчитися писати історії користувача, розробити історії користувача для ПС.</i> <i>Завдання:</i>	4	

<p>формальними методами опису вимог та моделювання.</p>	<p>1. Вивчити структуру написання історій користувача, 2. Визначити ролі в ПС, 3. Розробити історії користувача для ПС. <i>План заняття:</i> – Актуалізація теоретичного матеріалу – Виконання завдань практичної роботи – Презентація виконаної роботи</p>		
Тема 7. Процес управління змінами		15	13
<p><i>Загальні та спеціальні компетентності:</i> здатність до пошуку, оброблення та аналізу інформації з різних джерел; здатність формулювати та забезпечувати вимоги щодо якості програмного забезпечення відповідності з вимогами замовника, технічним завданням та стандартами. <i>Результати навчання:</i> знати та вміти використовувати методи та засоби збору, формулювання та аналізу вимог до програмного забезпечення.</p>	<p>Лекція 7. Процес управління змінами в традиційній та Agile розробці. <i>План лекції:</i> 1. Мета управління вимогами, 2. Відстеження вимог, 3. Прийоми дослідження вимог, 4. Аналіз здійсненності. Список рекомендованих джерел: Основний: 1-4, 8-13, 16 Додатковий: 6, 10 Інтернет-ресурси: 1-5, 11</p>	2	
<p>вибирати вихідні дані для проектування, керуючись формальними методами опису вимог та моделювання.</p>	<p>Завдання для самостійної роботи: Вивчення та доповнення матеріалів лекції на основі самостійного опрацювання основних літературних джерел, зазначених у списку на електронних матеріалів, які надаються викладачем Питання, внесені на самостійне опрацювання: Поняття про XML. 1. Поняття про мову XML, 2. Призначення XML, 3. Простий XML-документ.</p>	9	
<p>вибирати вихідні дані для проектування, керуючись формальними методами опису вимог та моделювання.</p>	<p>Практичне заняття № 7. Розроблення шаблону сценарію варіанта використання. <i>Мета:</i> навчитися розробляти сценарії варіантів використання та розробляти шаблон. <i>Завдання:</i> 1. Визначити: користувача, описати сценарій, продумати альтернативи, 2. Виділити параметри, 3. Розробити шаблон варіанту використання, 4. Користуючись шаблоном, розробити варіанти використання ПС. <i>План заняття:</i> – Актуалізація теоретичного матеріалу</p>	4	

	–Виконання завдань практичної роботи –Презентація виконаної роботи		
Тема 8. Пріоритезація вимог.		15	13
<p><i>Загальні та спеціальні компетентності:</i> здатність до пошуку, оброблення та аналізу інформації з різних джерел здатність формулювати та забезпечувати вимоги щодо якості програмного забезпечення у відповідності з вимогами замовника, технічним завданням та стандартами. здатність здійснювати процес інтеграції системи, застосовувати стандарти і процедури управління змінами для підтримки цілісності, загальної функціональності і надійності програмного забезпечення. <i>Результати навчання:</i> знати та вміти використовувати методи та засоби збору, формулювання та аналізу вимог до програмного забезпечення. вміти проводити розрахунок економічної ефективності програмних систем.</p>	<p>Лекція 8. Пріоритезація вимог. <i>План лекції:</i> 1. Мета пріоритезації вимог, 2. Техніки пріоритезації вимог: MOSCOW, модель кано, техніка story mapping, методологія kj, техніка пріоритезації feature buckets. Список рекомендованих джерел: Основний: 1, 12-17 Додатковий: 1, 2, 6, 9, 10 Інтернет-ресурси: 12, 13</p>	2	
	<p>Завдання для самостійної роботи: Вивчення та доповнення матеріалів лекції на основі самостійного опрацювання основних літературних джерел, зазначених у списку на електронних матеріалів, які надаються викладачем Питання, внесені на самостійне опрацювання: Методологія BPMN 1. Поняття методології BPMN; 2. Синтаксис BPMN; 3. Ознаки класифікації подій.</p>	9	
	<p>Практичне заняття № 8. Розроблення макетів екранів проекту (мокапи) <i>Мета: навчитися презентувати ПС замовнику на етапі проектування за допомогою мокапів</i> <i>Завдання:</i> 1. Створити блоки в прототипі макета, 2. Наповнити блоки вмістом, 3. Виконати структурування блоків, 4. Оформити блоки з урахуванням юзабіліті и ергономіки. <i>План заняття:</i> 5. Актуалізація теоретичного матеріалу, 6. Виконання завдань практичної роботи, 7. Презентація виконаної роботи.</p>	4	
Разом		120 годин/ 4 кредити	100 балів
Підсумковий контроль		Іспит	

Рекомендовані джерела інформації

1. Базова література

1. Вигерс, Карл И. Разработка требований к программному обеспечению. / Карл И. Вигерс, Джой Битти - БХВ-Петербург. 2016. – 736 с. <http://www.twirpx.com/file/1073169/>
2. Светлов Н. М. Информационные технологии управления проектами: Учебное пособие / Н.М. Светлов, Г.Н. Светлова. - 2 изд., перераб. и доп. - М.: НИЦ ИНФРА-М, 2015. - 232с. - <http://znanium.com/bookread.php?book=429103>
3. Технология разработки программного обеспечения: Учеб. пос. / Л.Г. Гагарина, Е.В. Кокорева, Б.Д. Виснадул; Под ред. проф. Л.Г. Гагариной - М.: ИД ФОРУМ: НИЦ Инфра-М, 2013. - 400 с. - <http://znanium.com/bookread2.php?book=389963>
4. Основы построения автоматизированных информационных систем: Учебник / В.А. Гвоздева, И.Ю. Лаврентьева. - М.: ИД ФОРУМ: НИЦ Инфра-М, 2013. - 320 с. URL: <http://znanium.com/bookread.php?book=392285>
5. Вендров, А.М. CASE-технологии. Современные методы и средства проектирования информационных систем / А.М. Вендров. – М.: Финансы и статистика, 1998. – 176 с.
6. Марка, Д.А. Методология структурного анализа и проектирования SADT / Д.А. Марка, К. МакГоуэн. – М.: МетаТехнология, 1993. – 243 с.
7. Калянов, Г.Н. CASE. Структурный системный анализ (автоматизация и применение) / Г.Н. Калянов. – М.: Лори, 1996. – с.
8. ГОСТ 19.701–90 (ИСО 5807–85). Единая система программной документации. Схемы алгоритмов, программ данных и систем. Условные обозначения и правила выполнения.
9. UML спецификация. – www.omg.com.
10. Буч, Г. Объектно-ориентированный анализ и проектирование с примерами приложений на C++ / Г. Буч. – М.: Бином, 2001. – 560 с.
11. Буч, Г. Язык UML. Руководство пользователя / Г. Буч, Дж. Рамбо, А. Якобсон. - СПб.: Питер, 2004. - 432 с.
12. Леоненков, А.В. Самоучитель UML 2 / А.В. Леоненков. – СПб.: БХВ - Петербург, 2007. – 576с.
13. Леоненков, А.В. Объектно-ориентированный анализ и проектирование с использованием UML / А.В. Леоненков. – www.intuit.ru.
14. Фаулер, М. Архитектура корпоративных программных приложений / М. Фаулер. – М.: Издательский дом «Вильямс», 2004. – 544 с.
15. Фаулер, М. UML. Основы. Третье издание. / М. Фаулер. – М.: Символ-Плюс, 2006. – 192 с.
16. Анисимов, В.В. Проектирование информационных систем. Часть 1. Структурный подход: конспект лекций / В.В. Анисимов. – Хабаровск: Изд-во ДВГУПС, 2005. – 112 с.

17. Анисимов, В.В. Проектирование информационных систем. Часть 2. Объектно-ориентированный подход: конспект лекций / В.В. Анисимов. – Хабаровск: Изд-во ДВГУПС, 2007. – 100 с.

2. Допоміжна література

1. Маклаков, С.В. ВРwin и ERwin. CASE-средства разработки информационных систем / С.В. Маклаков. – М.: ДИАЛОГ-МИФИ, 2001. – 304 с.

2. (ВРwin) Где? Зачем? Как? / В.И. Дубейковский. – М.: ДИАЛОГ-МИФИ, 2004. – 464 с.

3. Информационные системы: Учебное пособие / О.Л. Голицына, Н.В. Максимов, И.И. Попов. - 2-е изд. - М.: Форум: НИЦ ИНФРА-М, 2014. - 448 с.: ил.; 60x90 1/16. - (Высшее образование). (переплет) ISBN 978-5-91134-833-5, 1000 экз. <http://znanium.com/bookread.php?book=435900>

4. Информационные технологии и системы: Учеб. пособие / Е.Л. Федотова. - М.: ИД ФОРУМ: НИЦ Инфра-М, 2013. - 352 с.: ил.; 60x90 1/16. - (Высшее образование). (переплет) ISBN 978-5-8199-0376-6, 500 экз. <http://znanium.com/bookread.php?book=374014>

5. Академия Microsoft: Анализ требований к автоматизированным информационным системам: - <http://www.intuit.ru/studies/courses/2188/174/info>

6. Анализ требований к информационным системам Конспект лекций Маглинец Ю.А. – <http://ivan-shamaev.ru/wp-content/uploads/2013/06/Information-systems-analysis-and-requirements-analysis.pdf> Анализ требований по Вигерсу (2004).

7. Этапы сбора требований. - <http://iiba.ru/requirements-analysis/analysis-of-requirements-wiegers-2004/>

8. С.И. Клевцов анализ и формирование требований к программному обеспечению информационных систем сбора и обработки данных Учебное пособие - http://rtf.sfedu.ru!/mps/umk/strdsgn_ch1.pdf

9. Л.В. Щеголева А.Н. Кириленко проектирование информационной системы: структурный подход Учебное пособие для студентов вузов Петрозаводск Издательство ПетрГУ 2013 – 14 с.

10. Д. Леффингуэлл, Д. Уидриг, Принципы работы с требованиями к программному обеспечению. Унифицированный подход, Вильямс, 2002

3. Интернет-ресурси

1. Академия Microsoft: Анализ требований к автоматизированным информационным системам: <http://www.intuit.ru/studies/courses/2188/174/info>

2. АНАЛИЗ ТРЕБОВАНИЙ К ИНФОРМАЦИОННЫМ СИСТЕМАМ Конспект лекций Маглинец Ю.А. <http://ivan-shamaev.ru/wp-content/uploads/2013/06/Information-systems-analysis-and-requirements-analysis.pdf>

3. Анализ требований по Вигерсу (2004). Этапы сбора требований. <http://iiba.ru/requirements-analysis/analysis-of-requirements-wiegers-2004/> Вигерс К. Разработка требований к программному обеспечению <http://www.twirpx.com/file/1073169/>

4. С.И. Клевцов анализ и формирование требований к программному обеспечению информационных систем сбора и обработки данных Учебное пособие - http://rtf.sfedu.ru/!mps/umk/strdsngn_ch1.pdf
5. Проектирование информационных систем - https://sites.google.com/site/anisimovkhv/learning/pris/lecture/tema8/tema8_4
6. Стандартизация требований в Scrum-проектах: <https://tproger.ru/articles/scrum-standardization/>
7. Подготовительный этап разработки программного обеспечения - <https://habr.com/ru/post/256915/>
8. Методология моделирования IDEF3 для описания потоков работ (Work Flow Modeling): <https://intellect.icu/metodologiya-modelirovaniya-idef3-dlya-opisaniya-potokov-rabot-work-flow-modeling-7320>
<https://intellect.icu/metodologiya-modelirovaniya-idef3-dlya-opisaniya-potokov-rabot-work-flow-modeling-7320>
9. Диаграмма потоков данных: <https://studfile.net/preview/2915208/page:15/>
10. Синтаксис и основные понятия языка XML, создание валидных документов - <https://www.script-coding.com/XML.html>
11. Как составлять вариант использования - http://okiseleva.blogspot.com/2015/11/blog-post_86.html
12. Техники определения приоритетов задач для it-команд - <https://nikolaj-sarry.info/upravlenie-proektami/tehniki-opredeleniya-prioritetov-zadach-dlya-it-komand/>
13. Создание мокапов - <https://convertmonster.ru/blog/landing-page-blog/4-jetapa-sozdaniya-lendinga-v-balsamiq-mockups/>