

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Таныгин Максим Олегович

Должность: и.о. декана факультета фундаментальной и прикладной информатики

Дата подписания: 21.09.2023 12:41:32

Уникальный программный ключ:

65ab2aa0d384efe8480e6a4c688eddbc475e411a

МИНОБРАЗОВАНИЯ И НАУКИ  
РОССИИ

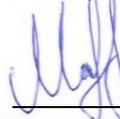
Юго-Западный государственный университет

УТВЕРЖДАЮ:

Заведующий кафедрой

программной инженерии

*(наименование кафедры полностью)*



А.В. Малышев

*(подпись)*

«28» апреля 2022 г.

## ОЦЕНОЧНЫЕ СРЕДСТВА

для текущего контроля успеваемости и

промежуточной аттестации

обучающихся по дисциплине

Разработка и анализ требований

*(наименование дисциплины)*

09.03.04 «Программная инженерия»

*(код)*

Направленность (профиль) «Разработка программно-информационных систем»

*и наименование ОПОП ВО)*

# 1 ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ

## 1.1 ВОПРОСЫ ДЛЯ СОБЕСЕДОВАНИЯ

1. *Понятие и классификация требований*
  - 1.1. Какие существуют классификации информационных систем?
  - 1.2. Что входит в состав требований?
  - 1.3. Что такое RUP?
  - 1.4. Какие методологии и стандарты регламентируют разработку требований?
2. *Требования и их свойства. Процесс анализа требований*
  - 2.1. Каким свойствам должны удовлетворять требования к разработке информационных систем?
  - 2.2. В чем заключается процесс анализа требований?
3. *Контекст задачи анализа требований. Выявление требований.*
  - 3.1. В чем заключается анализ предметной области?
  - 3.2. В чем заключается методология бизнес-анализа?
  - 3.3. Что является источником требований?
  - 3.4. Что такое прототипирование?
4. *Формирование видения. Специфицирование требований.*
  - 4.1. Каким образом определяются границы проекта?
  - 4.2. В чем смысл видения проекта?
  - 4.3. Что такое варианты использования информационной системы?
  - 4.4. Какие существуют шаблоны спецификации вариантов использования?
5. *Расширенный анализ требований. Моделирование и прототипирование.*
  - 5.1. Какие диаграммы UML можно использовать для расширенного анализа требований?
  - 5.2. В чем заключается смысл прототипирования?
  - 5.3. Какие имеются виды прототипов?
  - 5.4. Что такое раскадровка?
6. *Документирование и проверка требований.*
  - 6.1. Какие стандарты определяют требования к документированию требований к информационной системе?
  - 6.2. Какие существуют методы и средства проверки требований?
7. *Введение в управление требованиями.*
  - 7.1. В чем смысл базовой версии требований?
  - 7.2. Какие существуют процедуры управления требованиями?
  - 7.3. Как определяются трудозатраты?
  - 7.4. Каким образом осуществляется контроль изменений?
  - 7.5. В чем заключается смысл трассируемости требований?
8. *Совершенствование процессов работы с требованиями.*
  - 8.1. Какие существуют модели совершенствования?
  - 8.2. В чем заключается принцип совершенствования?
  - 8.3. Как производится оценка результатов принятия решений?
9. *Требования в управлении проектом.*
  - 9.1. В чем смысл экспресс-планирования?
  - 9.2. Что такое гибкие технологии проектирования информационных систем?
  - 9.3. Какие существуют стратегии выбора решений?

9.4. В чем заключается метод принятия решения на основании лучших практик?

**Шкала оценивания:** балльная.

**Критерии оценивания:**

**5-6 баллов** выставляется обучающемуся, если он демонстрирует глубокое знание содержания вопроса; дает точные определения основных понятий; аргументированно и логически стройно излагает учебный материал; иллюстрирует свой ответ актуальными примерами (типовыми и нестандартными), в том числе самостоятельно найденными; не нуждается в уточняющих и (или) дополнительных вопросах преподавателя.

**3-4 балла** выставляется обучающемуся, если он владеет содержанием вопроса, но допускает некоторые недочеты при ответе; допускает незначительные неточности при определении основных понятий; недостаточно аргументированно и (или) логически стройно излагает учебный материал; иллюстрирует свой ответ типовыми примерами.

**1-2 балла** выставляется обучающемуся, если он освоил основные положения контролируемой темы, но недостаточно четко дает определение основных понятий и дефиниций; затрудняется при ответах на дополнительные вопросы; приводит недостаточное количество примеров для иллюстрирования своего ответа; нуждается в уточняющих и (или) дополнительных вопросах преподавателя.

**0 баллов** выставляется обучающемуся, если он не владеет содержанием вопроса или допускает грубые ошибки; затрудняется дать основные определения; не может привести или приводит неправильные примеры; не отвечает на уточняющие и (или) дополнительные вопросы преподавателя или допускает при ответе на них грубые ошибки.

## **2 ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ**

### **2.1 БАНК ВОПРОСОВ И ЗАДАНИЙ В ТЕСТОВОЙ ФОРМЕ**

1. Вопросы в закрытой форме.

1.1. Какие из перечисленных ниже архитектурных стилей допускают понятия клиентского и серверного процессов?

Одноранговая архитектура.

Ярусная архитектура.

Архитектура, ориентированная на базы данных.

Ни одна из перечисленных.

1.2. Что измеряется при оценке усилий, требуемых для понимания программного обеспечения?

Структурная сложность.

Когнитивная сложность.

Алгоритмическая сложность.

Сложность задачи.

1.3. Шаблон, "обеспечивающий слабую связанность и избавляющий объекты от необходимости явно ссылаться друг на друга и независимо изменять способы взаимодействия".

Наблюдатель.

Фасад.

Абстрактная фабрика.

Ни один из перечисленных.

1.4. Какое из перечисленных ниже имен является синонимом шаблона Наблюдатель?

Слушатель.

Зритель.

Сторож.

Созерцатель.

1.5. Какой из перечисленных ниже объектов не может быть адресатом сообщения внутри класса в соответствии с законом Деметера?

Объект, на который ссылается глобальная переменная.

Объект, являющийся аргументом сигнатуры метода.

Объект, на который ссылается атрибут объекта ассоциированного класса.

Объект, созданный методом класса.

1.6. Что из перечисленного ниже не является артефактом, разрешенным для повторного использования?

Объект.

Класс.

Идея о решении.

Компонент.

1.7. Что является средством связи между ролями?

Ассоциации.

Соединительные звенья.

Связи.

Отношения.

1.8. Укажите, какая из перечисленных трудностей не представляет существенную сложность при разработке?

Согласованность.

Невидимость.

Корректность.

Изменяемость.

1.9. Что из перечисленного ниже относится к технологии OLAP?

Web-хранилище данных.

Хранилище данных.

Лавка данных.

Все вышеперечисленное.

1.10. Что из перечисленного ниже не относится к задачам интеллектуального анализа?

Классификация.

Кластеризация.

Упаковка.

Ни одно из перечисленных выше.

1.11. Что из перечисленного ниже не относится к методам структурного моделирования?

UML.

ERD.

DFD.

Все вышеперечисленное.

1.12. Что из перечисленного ниже не относится к методу итеративной и поступательной разработки?

Спиральная модель.

Функциональная декомпозиция.

Архитектура, ориентированная на модели.

Ничего из вышеперечисленного.

1.13. Какой из перечисленных ниже методов и моделей связан с программированием?

Аспектно-ориентированная разработка.

Ускоренная разработка программного обеспечения.

Генетическое программирование.

Ничего из вышеперечисленного.

1.14. Адаптивность программного обеспечения связана с одним из следующих аспектов.

Надежность.

Легкость использования.

Легкость сопровождения.

Все вместе.

1.15. Второй уровень технологической зрелости означает следующее.

Формализация и строгое следование процессам управления и разработки.

Для управления процессом используются количественные показатели.

Происходит непрерывное совершенствование процесса.

Ни одно из вышеперечисленных.

1.16. Программа непрерывного улучшения сервиса (Continuous Service Improvement Programme -- CSIP) является частью одного из следующих стандартов.

CMM.

ISO 9000.

ITIL.

COBIT.

1.17. Модель UML включает в себя следующие модели.

Модели состояний.

Модели изменения состояний.

Модели поведения.

Все вышеперечисленное.

1.18. Компоновка, ориентированная на процессы, является интеграцией элементов, обладающей следующими свойствами.

Она происходит на уровне баз данных или интерфейсов прикладного программирования (API), который предоставляет информацию другим приложениям.

В ней информация поступает из многих программных систем в общий пользовательский интерфейс.

Она связывает прикладные интерфейсы (т.е. сервисы, определенные с помощью абстракции интерфейса).

Ни одно из вышеперечисленных.

1.19. Что из перечисленного ниже не является методом планирования систем?

EPR.

SWOT.

ISA

Все.

1.20. Что из перечисленного ниже является основным видом деятельности в подходе VCM?

Управление кадрами.

Услуги.

Администрирование и обеспечение инфраструктуры.

Ни одно из вышеперечисленного.

1.21. Что из перечисленного ниже не относится к подходу ISA?

Субподрядчик.

Владелец.

Управление.

Все вышеперечисленное.

1.22. Что называется атомарным процессом в системе BPMN?

Действие.

Задача.

Событие.

Задание.

1.23. Что называется потоком сообщений в системе BPMN?

Потоковый объект.

Дорожка.

Артефакт.

Коннектор.

1.24. Что является элементом моделирования, определяющим бизнес-ценность функционального свойства в рамках концепции решения?

Бизнес-прецедент использования.

Прецедент решения.

Прецедент возможности.

Бизнес-прецедент.

1.25. Какой метод выявления требований работает с понятием провозможности?

Анкета.

"Мозговой штурм".

JAD.

Ни один из перечисленных.

1.26. Какой метод выявления требований работает с понятием триггерного вопроса?

Анкета.

RAD.

JAD.

Ни один из перечисленных.

1.27. Какое отношение утверждает, что один класс может быть разновидностью другого?

Обобщение.

Агрегация.

Ассоциация.

Ни один из перечисленных.

1.28. Что из перечисленного ниже является требованием к интерфейсу?

Функциональное требование.

Системный сервис.

Системное ограничение.

Ни одно из перечисленных.

1.29. Что такое MVC?

Модель.

Метамодель.

Среда программирования.

Все перечисленное выше.

1.30. Что из перечисленного ниже является именем яруса в архитектуре Core J2EE?

Бизнес.

Интеграция.

Представление.

Все перечисленное выше.

1.31. Что из перечисленного ниже является уровнем в архитектуре PCBMER, представляющем классы данных и объекты значений, предназначенные для отображения на экране дисплея?

Сущность.

Компонент.

Представление.

Ни один из перечисленных выше вариантов не подходит.

1.32. Как называется процесс отсылки уведомления объектам-подписчикам?

Обработка событий.

Пересылка.

Делегирование.

Ни один из перечисленных выше вариантов не подходит.

1.33. Какой из перечисленных ниже подходов к выявлению классов работает с концепцией нечеткого класса?

CRS.

Подход, основанный на общих шаблонных классах.

Подход, основанный на прецедентах использования.

Ни один из перечисленных выше вариантов не подходит.

1.34. В каких из категорий агрегаций наблюдаются свойства транзитивности и асимметрии?

Агрегация Owns.

Агрегация Has.

Агрегация ExclusiveOwns.

Все перечисленное выше.

1.35. Какое из перечисленных ниже отношений всегда связывает два прецедента использования?

Ассоциация.

Агрегация.

Расширение.

Все перечисленное выше.

1.36. Как называется событие, означающее, что сторожевое условие выполнено?

Изменение.

Временное событие.

Сигнал.

Вызов.

1.37. Действующее лицо -- это ...

Роль.

Сущность, внешняя по отношению к субъекту.

Сущность, действующая в прецеденте использования.

Все перечисленное выше.

1.38. Выполнением этапа деятельности называется...

Прецедент.

Действие.

Работа.

Поток.

1.39.

1.40. Параллельные потоки вычислений в диаграмме деятельности определяются с помощью...

Ветвления и воссоединения.

Разделения и слияния.

Сторожевых условий и слияния.

Все вышеприведенное не верно.

1.41. Классы, ответственные за взаимодействие с внешними источниками дан-МТ4. ных, называются...

Внешними классами.

Классами коммуникации.

Классами ресурсов.

Ни один вариант не подходит.

1.42. Черный ромб на линии агрегации означает, что...

Агрегация осуществляется по значению.

Агрегация осуществляется в композиции.

Объект подкласса может быть частью только одного суперкласса.

Все перечисленное выше.

1.43. Формальный аргумент может входить в определение...

Метода.

Сообщения.

Вызова.

Всего перечисленного выше.

1.44. Диаграмма конечного автомата -- это двудольный граф состояний и переходов, вызванных...

Деятельностью.

Триггерами.

Событиями.

Событиями.

1.45. Вычислительный ресурс, на котором может быть развернут артефакт для последующего выполнения, называется...

Компонентом.

Узлом.

Пакетом.

Все вышеприведенное не верно.

1.46. Что из перечисленного ниже не является механизмом расширения в языке UML?

Ограничение.

Стереотип.

Производный атрибут.

Меченое значение.

1.47. Что из перечисленного ниже является синонимом наследования интерфейса?

Выделение подтипа.

Заменимость.

Полиморфизм

Ничто из перечисленного выше.

1.48. Наследование, при котором происходит замещение унаследованных свойств МТЗ. в подклассе, называется...

Расширяющим наследованием.

Удобным наследованием.

Ограничивающим наследованием.

Все перечисленные варианты не верные.

1.49. В рамках какой концепции всегда появляется авторекурсия?

Делегирование.

Наследование интерфейса.

Пересылка.

Наследование реализации.

1.50. Как в языке UML 2.0 называется время, в течение которого поток управле-МТ5. ния фокусируется на объекте?

Использование взаимодействия.

Спецификация выполнения.

Линия жизни.

Все перечисленные варианты не верные.

1.51. Какой из следующих операторов определяет параллельный фрагмент, позволяющий перемежающееся выполнение вложенных функциональных свойств?

Opt.

Loop.

Alt.

Ни один из перечисленных выше.

1.52. Какой принцип проектирования графического пользовательского интерфейса можно связать с локализацией?

Индивидуализация.

Адаптивность.

Настройка.

Все перечисленные выше.

1.53. На какие части можно разделить окно графического пользовательского интерфейса?

Фреймы.

Формы.

Полосы.

Панели.

1.54. На какие части можно разделить окно графического пользовательского интерфейса Web-приложения?

Web-страницы.

Формы.

Фреймы.

Полосы.

1.55. С каким понятием связано понятие "песочницы"?

Web-приложения.

Настольные приложения.

Аплеты.

Страницы сервера.

1.56. Каким уровням архитектуры PCBMER принадлежат страницы JSP?

Компонентов.

Представления.

Управления.

Сущностей.

1.57. Какому уровню архитектуры PCBMER принадлежат действия форм Struts?

Компонентов.

Представления.

Управления.

Сущностей.

1.58. Что из перечисленного ниже не является стереотипом UX?

Окно.

Форма для ввода данных.

Экран.

Отделения.

1.59. К какой базе данных относится стандарт SQL:1999?

Реляционной.

Объектно-реляционной.

Объектно-ориентированной.

Ко всем вместе.

1.60. Что из перечисленного ниже не поддерживается моделью реляционных баз данных?

Структурные типы.

Ссылки.

Коллекции.

Ничего из выше перечисленного.

1.61. Для чего можно использовать представление?

Для программирования бизнес-правил.

Для поддержки безопасности базы данных.

Для определения доменов.

Ни для чего из перечисленного выше.

1.62. Какая стратегия не разрешается для отображения отношений обобщения?

Отображение всей иерархии классов в одну таблицу суперкласса.

Отображение каждого отдельного конкретного класса в таблицу.

Отображение каждого абстрактного класса в таблицу.

Допускаются все перечисленные выше стратегии.

1.63. Какой шаблон определен как "объект, содержащий не все данные, но знающий, где их найти"?

Единица работы.

Коллекция объектов.

Преобразователь данных.

Загрузка по требованию.

1.64. Какой вид блокировки допускает "грязное чтение"?

Блокировка намерения.

Блокировка чтения.

Разделяемая блокировка.

Ни одна из перечисленных выше.

1.65. Какой из перечисленных ниже методов относится к поддержке качества?

Сквозной контроль.

Контрольный список.

Ретроспекция.

Инспекция.

1.66. Выберите важную составляющую при разработке проекта, которая сокращает возможность проектной команды в выборе решений:

допущение

расширение

ограничение

оптимизация

- 1.67. Выберите ключевого участника программного проекта:
- спонсор
  - администратор
  - инициатор
  - ответственный
- 1.68. Выберите лицо или организацию, которые будут использовать продукт, услугу или результат проекта:
- заказчик
  - бухгалтер
  - директор
  - специалист
- 1.69. Выберите, представителя исполнителя, который уполномочен принимать решение о выделении ресурсов и изменениях в проекте:
- администратор
  - руководитель
  - спонсор
  - куратор
- 1.70. Выберите представителя исполнителя, ответственного за реализацию проекта в срок:
- администратор
  - заказчик
  - директор
  - руководитель
- 1.71. Выберите, что может отвечать на вопрос, зачем данный проект нужен?
- бизнес-план
  - смета
  - задачи
  - цель
- 1.72. Определите, какой документ разрабатывается на основе анализа потребностей:
- смета
  - бюджет
  - стратегия
  - концепция
- 1.73. Определите особенность при разработке проекта, которая тесно связана с управлением рисками:
- бюджетирование
  - оптимизация
  - допущения
  - ограничения
- 1.74. Как называются вопросы, на которые должен ответить испытатель?
- Точки верификации.
- Запросы сценария.
- Тестовые запросы.
- Все ответы неправильные.

1.75. Как называется повторение приемочных испытаний последовательных версий кода?

Тестирование по методу "прозрачного ящика".

Регрессионное тестирование.

Тестирование по методу "черного ящика".

Тестирование покрытия.

1.76. Как называется тестирование по методу "черного ящика"?

Тестирование по спецификациям. Функциональное тестирование.

Тестирование на основе ввода-вывода.

Все ответы неправильные.

1.77. Как называется тестирование по методу "прозрачного ящика"?

Функциональное тестирование.

Тестирование по спецификациям.

Тестирование ветвей программы.

Тестирование покрытия.

1.78. Выберите НЕ существующую фазу жизненного цикла проекта:

Инициирование

Планирование

Реализация

Мониторинг

Завершение

1.79. Для эффективной работы с внешними участниками проекта требуется:

выявить всех участников

выявить участников проекта на начальной фазе

провести классификацию участников по степени их воздействия на проект

установить приоритеты в требуемых связях с внешними участниками

установить приоритеты в требуемых связях с внутренними участниками

провести классификацию участников по степени их успеваемости

2. Вопросы в открытой форме.

2.1. \_\_\_\_\_ - требования описывают ожидаемое поведение системы при определенных условиях и действия, которые система позволит выполнять пользователям.

2.2. Если клиент заявляет, что только определенные классы пользователей могут выполнять определенные действия при определенных условиях, он, возможно, описывает \_\_\_\_\_

2.3. Вся информация, описывающая финансовые, рыночные или другие отношения коммерческого характера, которые клиенты или компания-разработчик собираются получить от использования продукта, относится к \_\_\_\_\_

2.4. Утверждения, насколько хорошо система выполняет что-то, называются \_\_\_\_\_

2.5. Сущность \_\_\_\_\_ состоит в задании целей и способов их достижения на основе формирования комплекса работ (мероприятий, действий), которые должны быть выполнены, применении методов и средств реализации этих работ, увязки ресурсов, необходимых для их выполнения, согласовании действий организаций-участников проекта

2.6. \_\_\_\_\_ это описание одного случая использования системы.

- 2.7. \_\_\_\_\_ определяет, какие операции не может выполнять система и ее пользователи
- 2.8. \_\_\_\_\_ - группа, охватывающая тех людей, кто будет непосредственно использовать программное обеспечение; пользователи могут описать задачи, которые они решают (планируют решать) с использованием программной системы, а также ожидания по отношению к атрибутам качества, отображаемые в пользовательских требованиях.
- 2.9. \_\_\_\_\_ - те, кто отвечают за заказ программного обеспечения или, в общем случае, являются целевой аудиторией на рынке программного обеспечения (образуют целевой рынок ПО)
- 2.10. \_\_\_\_\_ – контекст для сбора пользовательских требований, определяющий ответы на вопросы “что если” и “как это делается” в отношении бизнес-процессов, реализуемых пользователями;.
- 2.11. \_\_\_\_\_ - непосредственное присутствие аналитиков и инженеров рядом с пользователем в процессе выполнения последним его работ по обеспечению функционирования бизнес-процессов
- 2.12. \_\_\_\_\_ тестирование - тестируется интегрированная система на её соответствие исходным требованиям.
- 2.13. \_\_\_\_\_ предприятия - исходя из стратегии организации определяет цели и основные требования к проекту.
- 2.14. Можно говорить о том, что процедура \_\_\_\_\_ требований считается выполненной только тогда, когда все требования, включенные в спецификацию, обладают методами оценки соответствия им создаваемого программного продукта.

### 3. Вопросы на установление последовательности.

#### 3.1. Укажите очередность шагов процесса создания требований:

- Определите классы пользователей
- Определите представителей пользователей
- Определите бизнес-требования
- Спланируйте выявление требований
- Определите пользовательские требования
- Определите приоритеты пользовательских требований
- Определите лиц, ответственных за принятие решений по требованиям

#### 3.2. Последовательность шагов при создании инфраструктуры взаимодействия по А.Куперу.

- Определение форм-фактора, типа приложения и способов управления.
- Определение функциональных и информационных элементов.
- Выполнение проверочных сценариев для верификации решений.
- Макетирование общей инфраструктуры взаимодействия.
- Создание ключевых сценариев.
- Определение функциональных групп и иерархических связей между ними.

#### 3.3. Последовательность шагов при выработке требований с использованием персонажей и сценариев

- Выявление ожиданий персонажей.
- Постановка задач и определение образа продукта.
- Выявление требований.

Мозговой штурм.

Разработка контекстных сценариев.

#### 4. Вопросы на установление соответствия.

##### 4.1. Установите соответствие

1. Бизнес-требования (Business Requirements) –
2. Пользовательские требования (User Requirements) –
3. Функциональные требования (Functional Requirements) –

а) описывают цели/задачи пользователей системы, которые должны достигаться/выполняться пользователями при помощи создаваемой программной системы. Эти требования часто представляют в виде вариантов использования (Use Cases).

б) определяют функциональность (поведение) программной системы, которая должна быть создана разработчиками для предоставления возможности выполнения пользователями своих обязанностей в рамках бизнес-требований и в контексте пользовательских требований.

с) определяют высокоуровневые цели организации или клиента (потребителя) – заказчика разрабатываемого программного обеспечения.

##### 4.2. Установите соответствие

1. Группа нефункциональных требований
  2. Группа функциональных требований
- а) Бизнес-требования (Business Requirements)
  - б) Пользовательские требования (User Requirements)
  - с) Бизнес-правила (Business Rules)
  - д) Внешние интерфейсы (External Interfaces)
  - е) Атрибуты качества (Quality Attributes)
  - ф) Ограничения (Constraints)
  - г) Системные требования (System Requirements)

##### 4.3. Установите соответствие

1. Спецификация требований.
  2. Утверждение требований.
- а) Документ определения системы
  - б) Обзор требований

##### 4.4. Установите соответствие

1. Спецификация требований.
  2. Утверждение требований.
  3. Извлечение требований.
- а) Источники требований
  - б) Приемочные тесты
  - в) Документ определения системы

**Шкала оценивания результатов тестирования:** в соответствии с действующей в университете балльно-рейтинговой системой оценивание результатов промежуточной аттестации обучающихся осуществляется в рамках 100-балльной шкалы, при этом максимальный балл по промежуточной аттестации обучающихся по очной форме обучения

составляет 36 баллов, по очно-заочной и заочной формам обучения – 60 баллов (установлено положением П 02.016).

Максимальный балл за тестирование представляет собой разность двух чисел: максимального балла по промежуточной аттестации для данной формы обучения (36 или 60)

и максимального балла за решение компетентностно-ориентированной задачи (6).

Балл, полученный обучающимся за тестирование, суммируется с баллом, выставленным ему за решение компетентностно-ориентированной задачи.

Общий балл по промежуточной аттестации суммируется с баллами, полученными обучающимся по результатам текущего контроля успеваемости в течение семестра; сумма баллов переводится в оценку по *дихотомической* шкале:

Соответствие 100-балльной и дихотомической шкал

*Сумма баллов по 100-балльной шкале*

*Оценка по дихотомической шкале*

100-50

зачтено

49 и менее

не зачтено

### **2.3 КОМПЕТЕНТНОСТНО-ОРИЕНТИРОВАННЫЕ**

*Компетентностно-ориентированная задача № 1*

Составить пользовательские истории для разработки требований к сайту университета.

*Компетентностно-ориентированная задача № 2* Составить пользовательские истории для разработки требований к сайту школы.

*Компетентностно-ориентированная задача № 3* Составить пользовательские истории для разработки требований к мобильному приложению для заказа товаров.

*Компетентностно-ориентированная задача № 4* Составить пользовательские истории для разработки требований к мобильному приложению для заказа такси.

*Компетентностно-ориентированная задача № 5*

Составить пользовательские истории для разработки требований к мобильному приложению для обучения.

*Компетентностно-ориентированная задача № 6*

Разработать техническую спецификацию на программный компонент разработки сайта университета.

*Компетентностно-ориентированная задача № 7*

Разработать техническую спецификацию на программный компонент разработки сайта школы.

*Компетентностно-ориентированная задача № 8*

Составить требования для интерфейса для сайта университета.

*Компетентностно-ориентированная задача № 9*

Составить требования для интерфейса для сайта школы.

*Компетентностно-ориентированная задача № 10*

Составить требования для интерфейса для мобильного приложения для обучения.

*Компетентностно-ориентированная задача № 11*

Разработать требования к тестированию модуля для ввода персональной информации работника корпоративной ИС.

*Компетентностно-ориентированная задача № 12*

Разработать требования к тестированию модуля для ввода персональной информации в систему заказа билетов.

*Компетентностно-ориентированная задача № 13*

Осуществить декомпозицию работ по разработке сайта интернет-магазина и определить сроки разработки каждой работы .

*Компетентностно-ориентированная задача № 14*

Осуществить декомпозицию работ по разработке мобильного приложения заказа такси и определить сроки разработки каждой работы.

**Шкала оценивания решения компетентностно-ориентированной задачи:** в соответствии с действующей в университете балльно-рейтинговой системой оценивание результатов промежуточной аттестации обучающихся осуществляется в рамках 100-балльной шкалы, при этом максимальный балл по промежуточной аттестации обучающихся по очной форме обучения составляет 36 баллов, по очно-заочной и заочной формам обучения – 60 (установлено положением П 02.016).

Максимальное количество баллов за решение компетентностно-ориентированной задачи – 6 баллов.

Балл, полученный обучающимся за решение компетентностно-ориентированной задачи, суммируется с баллом, выставленным ему по результатам тестирования.

Общий балл промежуточной аттестации суммируется с баллами, полученными обучающимся по результатам текущего контроля успеваемости в течение семестра; сумма баллов переводится в оценку по *дихотомической* шкале.

Соответствие 100-балльной и дихотомической шкал

*Сумма баллов по 100-балльной шкале*

*Оценка по дихотомической шкале*

100-50

зачтено

49 и менее

не зачтено

**Критерии оценивания решения компетентностно-ориентированной задачи** (нижеследующие критерии оценки являются примерными и могут корректироваться):

**6-5 баллов** выставляется обучающемуся, если решение задачи демонстрирует глубокое понимание обучающимся предложенной проблемы и разностороннее ее рассмотрение; свободно конструируемая работа представляет собой логичное, ясное и при этом краткое, точное описание хода решения задачи (последовательности (или выполнения) необходимых трудовых действий) и формулировку доказанного, правильного вывода (ответа); при этом обучающимся предложено несколько вариантов

решения или оригинальное, нестандартное решение (или наиболее эффективное, или наиболее рациональное, или оптимальное, или единственно правильное решение); задача решена в установленное преподавателем время или с опережением времени.

**4-3 балла** выставляется обучающемуся, если решение задачи демонстрирует понимание обучающимся предложенной проблемы; задача решена типовым способом в установленное преподавателем время; имеют место общие фразы и (или) несущественные недочеты в описании хода решения и (или) вывода (ответа).

**2-1 балла** выставляется обучающемуся, если решение задачи демонстрирует поверхностное понимание обучающимся предложенной проблемы; осуществлена попытка шаблонного решения задачи, но при ее решении допущены ошибки и (или) превышено установленное преподавателем время.

**0 баллов** выставляется обучающемуся, если решение задачи демонстрирует непонимание обучающимся предложенной проблемы, и (или) значительное место занимают общие фразы и голословные рассуждения, и (или) задача не решена.