|  |
| --- |
| **К ЭКЗАМЕНУ ДОПУСКАЮТСЯ ВСЕ !!! (кроме недопущенных по курсовым, оплате и двум несданным экзаменам до того )** |
|  |
| **Первые вопросы в билете (теоретические вопросы)** |
| Жизненный цикл программного обеспечения |
| Каскадная модель разработки ПО |
| Спиральная модель разработки ПО |
| Эволюционное прототипирование |
| Рациональный унифицированный процесс |
| Гибкая разработка (манифест, принципы, практики) |
| Scrum: роли |
| Scrum: артефакты |
| Scrum: встречи |
| Экстремальное программирование: ценности и принципы |
| Уровни требований к ПО |
| Бизнес-требования и требования пользователей |
| Функциональные и нефункциональные требования |
| Характеристики качества требований |
| Атрибуты качества ПО |
| Основные элементы языка UML |
| Типы сущностей UML |
| Основные отношения в UML и их модификации |
| Диаграммы UML (виды, назначение, элементы) |
| Механизмы расширения языка UML |
| Основные понятия в тестировании ПО |
| Верификация и валидация |
| Виды тестирования ПО |
| Ручное и автоматизированное тестирование |
| Подготовка тестовых данных и проектирование тест-кейсов |
| Метрики в тестировании ПО |
| Организация тестирования ПО |
| Разработка управляемая тестированием |
| Особености тестирования веб-приложений |
| Особенности тестирования мобильных приложений |
|  |
| **Вторые вопросы в билете (паттерны проектирования и шаблоны рефакторинга)** |
|  |
| ***Паттерны проектирования (назначение, классификация, применение, примеры использования; для каждого паттерна построить UML диаграммы классов и последовательности)*** |
| Паттерн проектирования Наблюдатель (Observer) |
| Паттерн проектирования Итератор (Iterator) |
| Паттерн проектирования Стратегия (Strategy) |
| Паттерн проектирования Состояние (State) |
| Паттерн проектирования Адаптер (Adapter) |
| Паттерн проектирования Заместитель (Proxy) |
| Паттерн проектирования Композит (Composite) |
| Паттерн проектирования Фасад (Facade) |
| Паттерн проектирования Абстрактная фабрика (Abstract Factory) |
| Паттерн проектирования Фабричный метод (Factory Method) |
| Паттерн проектирования Одиночка (Singleton) |
| Паттерн проектирования Хранитель (Memento) |
| Паттерн проектирования Посетитель (Visitor) |
| Паттерн проектирования Декоратор (Decorator) |
| Паттерн проектирования по выбору\* |
|  |
| ***Шаблоны рефакторинга (назначение, классификация, последствия, примеры, обязательно проиллюстрировать UML диаграммами классов и последовательности)*** |
| Рефакторинг Выделение метода (Extract method) |
| Рефакторинг Перемещение метода (Move method) |
| Рефакторинг Перемещение поля (Move field) |
| Рефакторинг Выделение класса (Extract class) |
| Рефакторинг Замена значения данных объектом (Replace data value with object) |
| Рефакторинг Дублирование видимых данных (Duplicate observed data) |
| Рефакторинг Замена кода типа состоянием (Replace type code with state) |
| Рефакторинг Замена кода типа стратегией (Replace type code with strategy) |
| Рефакторинг Подъём поля (Pull up field) |
| Рефакторинг Подъём метода (Pull up method) |
| Рефакторинг Выделение родительского класса (Extract superclass) |
| Рефакторинг Выделение интерфейса (Extract interface) |
| Рефакторинг Замена конструктора фабричным методом (Replace constructor with factory method) |
| Рефакторинг Замена условного оператора полиморфизмом (Replace conditional with polymorphism) |
| Рефакторинг по выбору\* |
|  |
| **Третьи вопросы в билете (практические задания)** |
| ***Приёмы работы с Git (из консоли, можно использовать свой репозиторий)*** |
| Основные приёмы работы с Git (конфигурация, создание репозитория, клонирование) |
| Основные приёмы работы с Git (работа с коммитом: добавление/удаление списков файлов, работа с локальным репозиторием) |
| Основные приёмы работы с Git (работа с внешним репозиторием: получение/отправка данных, ветвление) |
| Основные приёмы работы с Git (тэги, игнорирование файлов) |
| Основные приёмы работы с Git (reset и revert) |
| ***Приёмы работы с IntelliJ IDEA (продемонстировать на примере работы со своим проектом)*** |
| Основные приёмы работы с IntelliJ IDEA (работа с редактором кода) |
| Основные приёмы работы с IntelliJ IDEA (структура и сборка проекта) |
| Основные приёмы работы с IntelliJ IDEA (рефакторинг кода) |
| Основные приёмы работы с IntelliJ IDEA (работа с JUnit) |
| Основные приёмы работы с IntelliJ IDEA (работа с Git) |
| ***Постороение диаграмм UML в Rational Rose*** |
| Построение UML диаграммы: Вариантов использования (Use Case) |
| Построение UML диаграммы: Классов (Class) |
| Построение UML диаграммы: Последовательности (Sequence) |
| Построение UML диаграммы: Состояний (State chart) |
| Построение UML диаграммы: Коопераций (Collaboration) |
| Построение UML диаграммы: Активностей (Activity) |
| Построение UML диаграммы: Компонентов (Component) |
| Построение UML диаграммы: Развёртывания (Deployment) |
| Построение UML диаграммы: Объектов (Object) |
| Построение UML диаграммы: Пакетов (Package) |
| ***Тестирование одного из разделов сайта, объект тестирования - по билету, спроектировать необходимые тест кейсы (позитивные и негативные) и продемонстрировать их выполнение, ошибки*** |
| Функциональное тестирование сайта bsuir.by (раздел “Расписание”) |
| Юзабилити тестирование сайта bsuir.by (раздел “Расписание”) |
| Тестирование надёжности сайта bsuir.by (раздел “Расписание”) |
| Тестирование безопасности сайта bsuir.by (раздел “Расписание”) |
| Тестирование переносимости сайта bsuir.by (раздел “Расписание”) |
| Функциональное тестирование сайта bsuir.by (раздел “Поиск”) |
| Юзабилити тестирование сайта bsuir.by (раздел “Поиск”) |
| Функциональное тестирование сайта bsuir.by (раздел “Телефонный справочник”) |
| Юзабилити тестирование сайта bsuir.by (раздел “Телефонный справочник”) |
| Юзабилити тестирование сайта bsuir.by (Главная страница) |