**Вопросы к промежуточной аттестации**

**по дисциплине «Коллективная разработка ИС»:**

1. Понятие системы контроля версий. Распределенные системы контроля версий. Примеры.

2. Репозиторий Git. Сервисы хостинга репозитория Git. Команды работы с удаленным репозиторием.

3. Понятие Git. Преимущества и недостатки использования.

4. Откат изменений кода. Команды Discard, Revert, Reset.

5. Работа с веткамиы.

6. Ветвление и и слияние. Основные команды.

7. Разрешение конфликтных ситуаций.

8. Хранение распределенный файлов в репозитории.

9. Pull Request на примере работы с Open-Source-проектами

10. Консольный git-клиент и интерфейсы

11. Принципы и значение гибкой методологии разработки

12. Agile. Agile Manifesto.

13. Процессный подход. Моделирование, анализ, автоматизация и контроль бизнес-процессов.

14. Scrum. Роли Scrum.

15. Этапы и мероприятия Scrum.

16. Атрибуты Scrum.

17. Техники и методики оценки Scrum-работ.

18. Kanban.

19. Lean.

20. Экстремальное программирование.

21. RUP.

22. Crystal Clear.

23. Waterfall.

1. ATDD. Разработка через приемочное тестирование.
2. Гибкая методология разработки. Основные принципы методологии Agile.
3. Экстремальное программирование. Основные принципы методологии XP.
4. Методология SCRUM. Основные понятия, Спринты, роли, собрания.
5. Системы совместной разработки ПО. Системы управления репозиториями.
6. Сравнение систем контроля версий Git и Mercurial.
7. Методология MSF. Версии. Модель проектной группы. Роли в проектной группе.
8. Сравнение применения технологий MSF, RUP и XP при коллективной разработке ПО
9. Использование методологии RUP для коллективной разработке ИС
10. Программное обеспечение с открытым исходным кодом. Преимущества подхода.
11. Централизованные и децентрализованные системы коллективной разработки. Основные понятия и анализ.
12. Дополнительные средства коллективной разработки. Wiki, трекеры. Основные понятия, обзор.
13. Технические командные роли при коллективной разработке ИС
14. Модели коллективной разработки ИС
15. Технологии коллективной разработки.
16. Психологические командные роли при коллективной разработке ИС
17. Повышение эффективности коллективной работы
18. Концепция парного программирования. Преимущества и недостатки.
19. Системы контроля версий - основа командной разработки. Распределенные системы контроля версий.
20. Типы совместной деятельности при коллективной разработке (мандатная, созываемая, естественная, индивидуальная).
21. Формирование команды ИТ-проекта. Ключевые роли, их назначения и взаимодействие на проекте
22. Модель группы и иерархическая модель. Обязанности членов группы.
23. Коллективная разработка ПО. Размеры группы и масштаб проекта.
24. Иерархическая модель организации коллектива при разработке ПО. Преимущества и недостатки.
25. Преимущества и недостатки применения систем контроля версий при коллективной разработке ИС

**Типовые практические задания к промежуточной аттестации.**

1. Написать команду слияния двух веток в git

2. Создать конфликт и разрешить его в git

3. Создать новую ветку в git

4. Добавить комментарий к файлу в git

5. Выполнить откат изменений кода в git

6. Добавить данные в удаленный репозиторий

7. Выполнить «прятанье» файла в git

8. Выполнить перемещение файлов в в git

9. Выполнить сравнение файлов в git

10. Выполнить просмотр истории коммитов