

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

Дисциплина: Фронт-энд разработка

Отчет

Домашняя работа

Выполнил:

Смирнов Александр

Группа

Проверил:

Добряков Д. И.

Санкт-Петербург

2026 г.

Задача

Было необходимо пройти три браузерные игры:

1. flexboxfroggy
2. cssgridgarden
3. learnitbranching

Ход работы

Выполнял задания в играх поочередно, изучал теоретические справки, затем выполнял теорию. К подсказкам обращался только в learnitbranching в паре сложных уровней, все остальное решал самостоятельно.

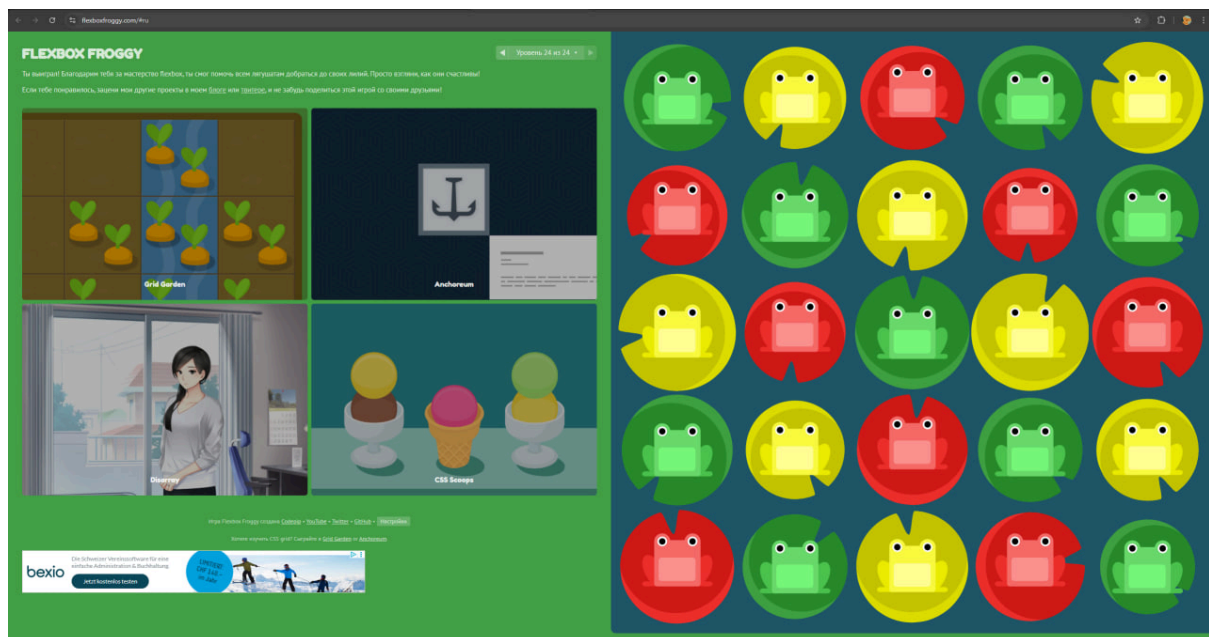


Рис. 1 Финальный экран flexboxfroggy

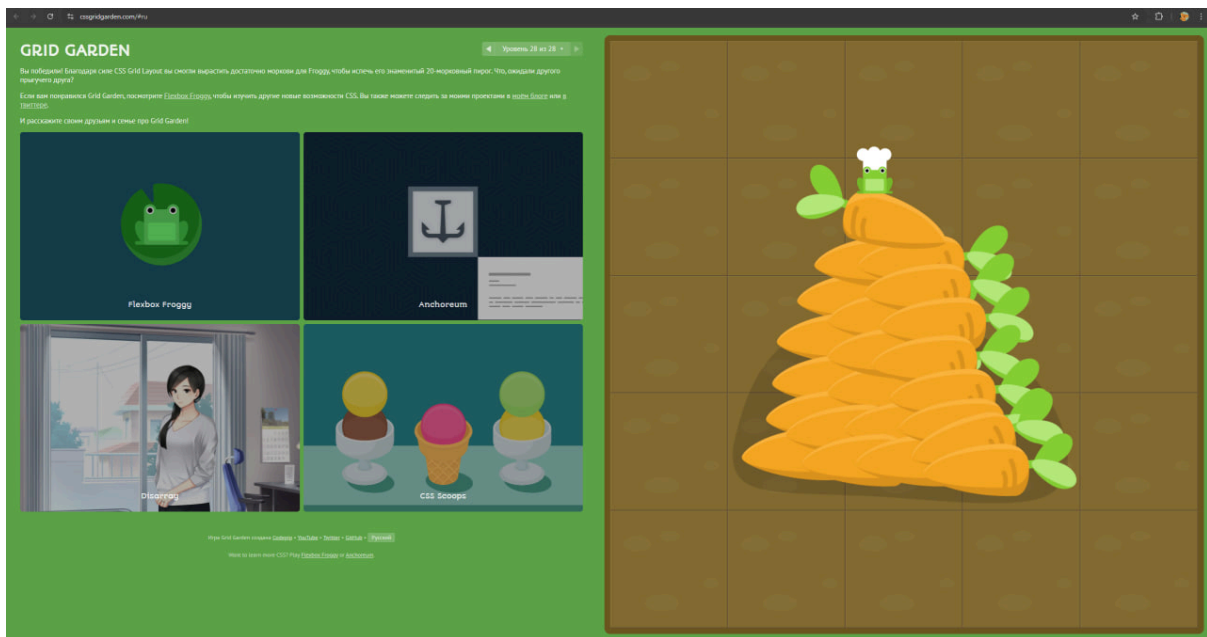


Рис. 2 Финальный экран cssgridgarden

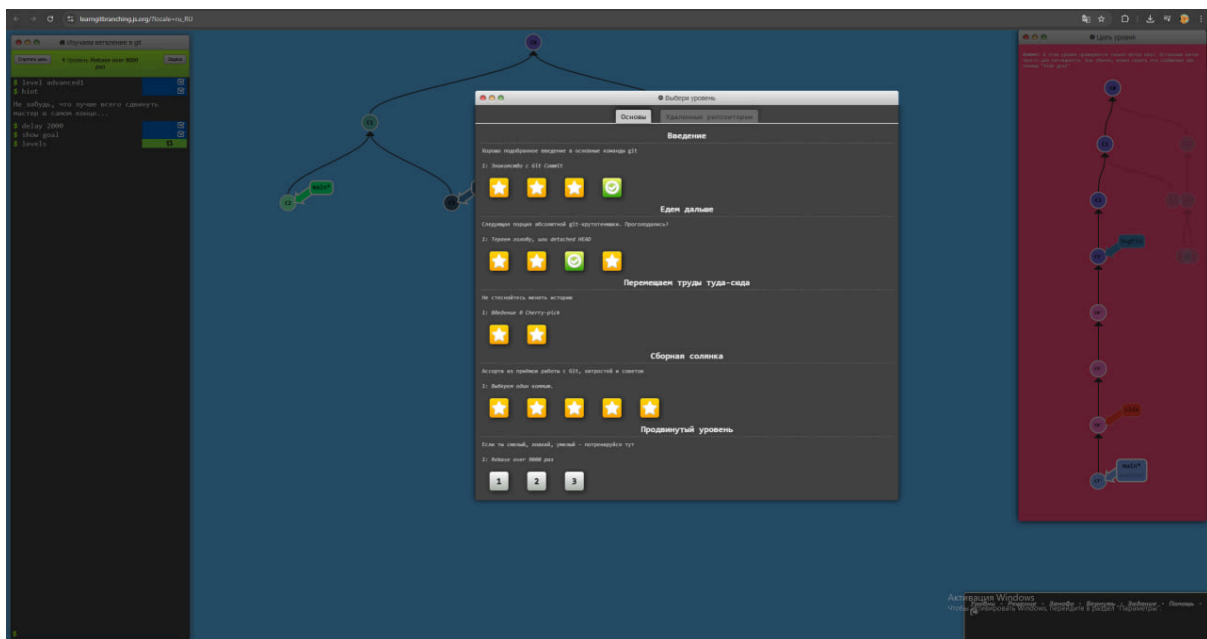


Рис. 3 Пройденные уровни по основам Git

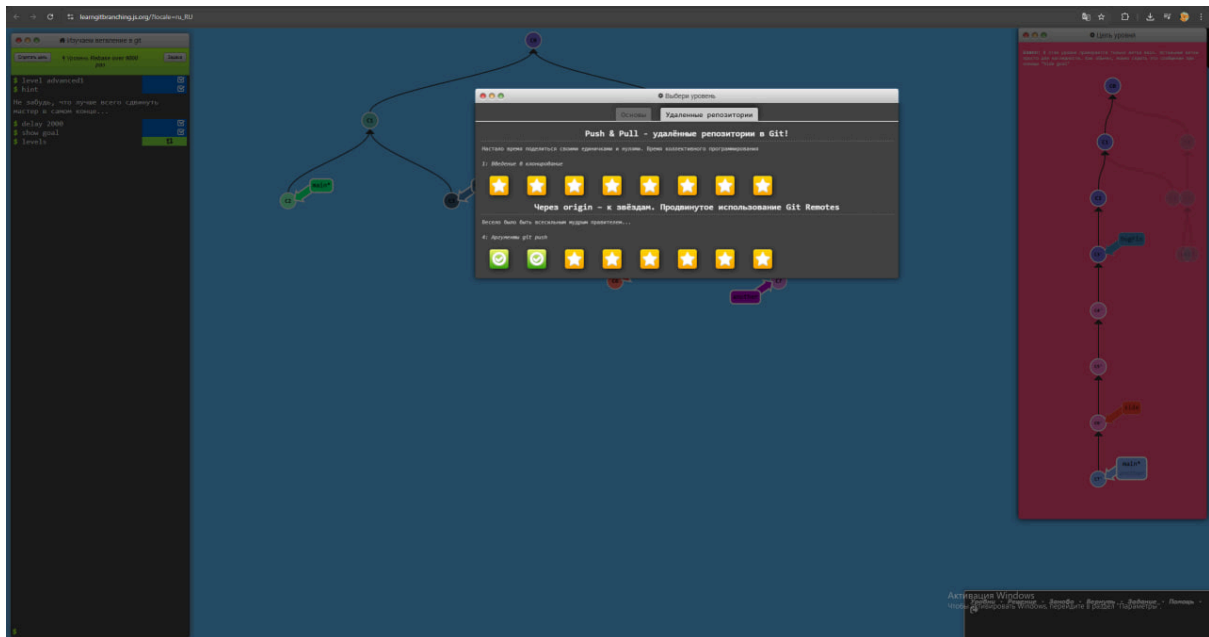


Рис. 4 Пройденные уровни по удаленным репозиториям Git

Вывод

1. Flexbox Froggy (CSS Flexbox)

В ходе выполнения заданий были отработаны следующие свойства и принципы:

- flex-direction - определение направления основной оси контейнера.
- justify-content - выравнивание элементов вдоль главной оси.
- align-items - выравнивание элементов вдоль поперечной оси.
- flex-wrap - управление переносом элементов на новую строку при недостатке места.
- align-content - выравнивание строк внутри контейнера при наличии свободного пространства.
- order - изменение визуального порядка отдельных элементов.

2. CSS Grid Garden (CSS Grid Layout)

- В процессе прохождения тренажёра были изучены и закреплены следующие аспекты грид-вёрстки:
- grid-template-columns и grid-template-rows - задание структуры и размеров столбцов и строк сетки.

- `grid-template` - сокращённая форма для одновременного определения столбцов и строк.
- `grid-column` и `grid-row` - позиционирование элементов по линиям сетки, включая использование ключевого слова `span` для указания ширины или высоты элемента.
- `grid-area` - определение области элемента либо через указание координат, либо через именованные зоны.

3. *Git Branching (Практикум по системе контроля версий)*

- Данный тренажёр был направлен на углублённое понимание модели ветвления в Git. В процессе выполнения заданий были рассмотрены следующие аспекты:
- Локальная работа с ветками: создание и переключение между ветками, выполнение слияний, разрешение конфликтов. Отдельное внимание уделено механизму перемещения указателей HEAD и работе с относительными ссылками.
- Инструменты для работы с историей: освоены команды интерактивного изменения истории, включая `rebase -i`, а также операция `cherry-pick` для выборочного переноса изменений.
- Удалённое взаимодействие: рассмотрены сценарии синхронизации с удалёнными репозиториями, включая особенности работы с отслеживаемыми ветками и разрешение ситуаций с расходящейся историей.

Ранее в работе с Git я пользовался только базовыми операциями (`commit`, `push`, `pull`) и простым слиянием. Прохождение данного тренажёра позволило выйти за рамки повседневных сценариев и освоить более тонкие инструменты управления историей проекта. В частности, практика с интерактивным `rebase` дала понимание того, как можно поддерживать чистоту и линейность истории коммитов перед интеграцией в основную ветку разработки.