# Assignment #3 — Advanced CSS: Flexbox & Grid

Team: Vitality  
Members: Sariyev Maxot, Tanibergen Beknur  
Group: SE-2406  
Deployed URL:

## Objective

Implement modern responsive layouts using Flexbox and CSS Grid across the project’s pages, focusing on alignment, spacing, equal heights, grid areas, and hover interactions.

## Pages and Files

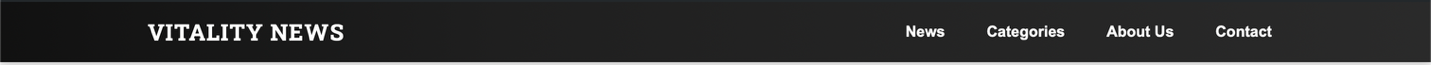
* index.html — homepage
* html/news.html — news grid with articles
* html/categories.html — categories + gallery
* html/about.html — about page
* html/contact.html — contact page
* styles/style.css — shared styles (header, nav, hero, cards)
* styles/news.css — news page grid layout and cards
* styles/categories.css — categories and gallery

## Part 1 — Flexbox

### Task 1: Navigation Bar

* Header uses Flexbox for horizontal alignment of logo and links.
* Vertically centered with spacing between links.

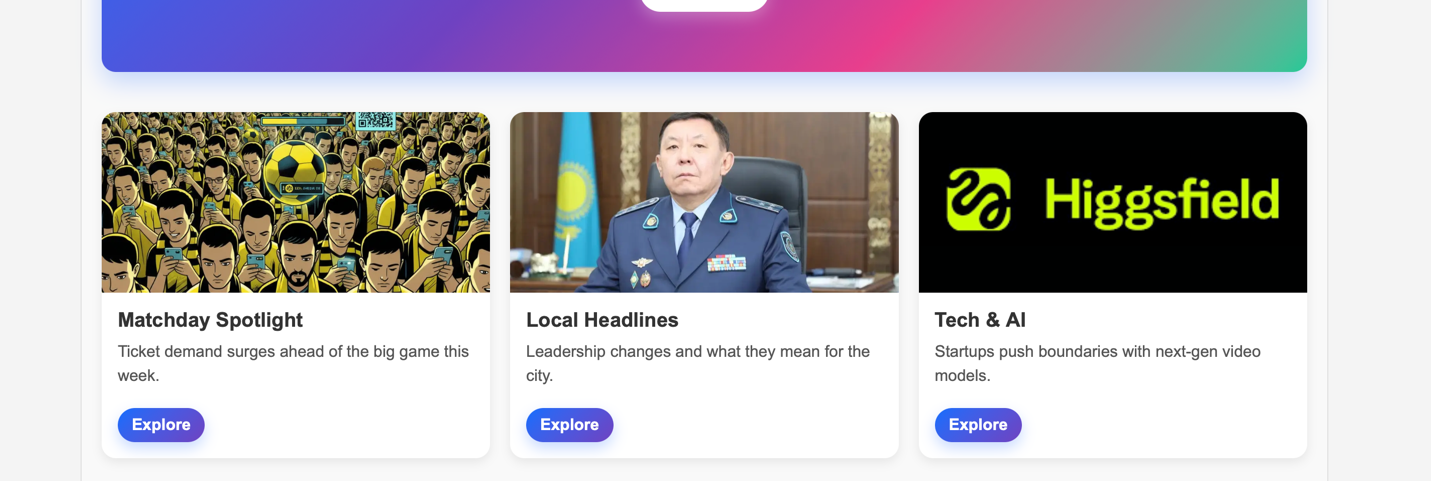
Code references: - styles/style.css - nav { display: flex; justify-content: space-between; align-items: center; } - nav ul { display: flex; } - nav ul li a { padding, border-radius, hover background }



### Task 2: Card Row (Homepage)

* Added 3 cards in index.html under <section class="card-row">.
* Container card-row is Flexbox with wrapping and gaps.
* Cards have equal height via column layout and flexible body area.
* Hover effects: elevation and slight translateY.

Key styles in styles/style.css: - .card-row { display: flex; flex-wrap: wrap; gap: 20px; align-items: stretch; } - .card { display: flex; flex-direction: column; } - .card-body { flex: 1 1 auto; }

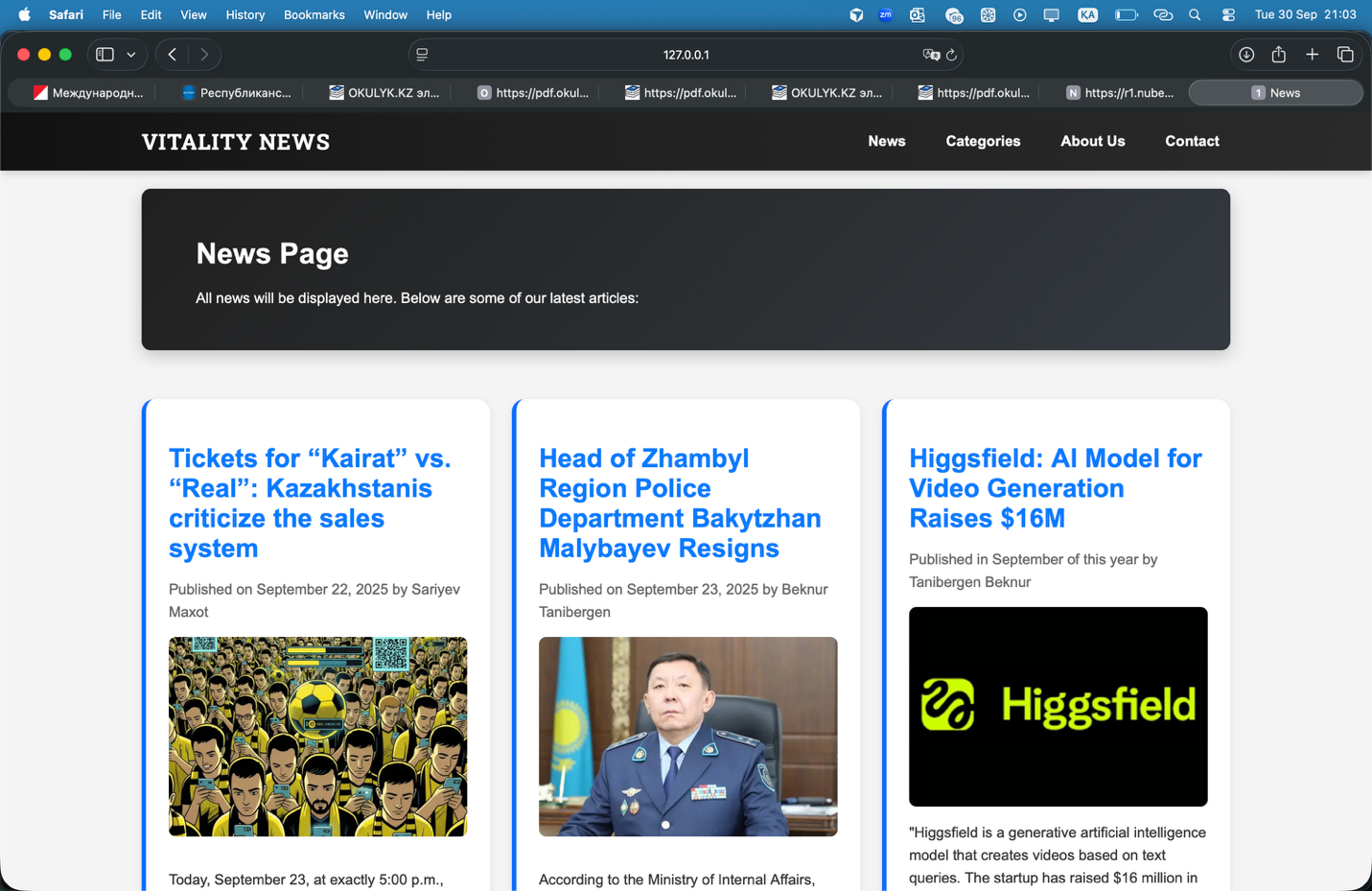


## Part 2 — Grid System

### Task 3: Page Layout with Grid Areas (News Page)

* The main on html/news.html uses a single-column grid with named areas: header, main, footer.
* The grid supports responsive behavior without external frameworks.

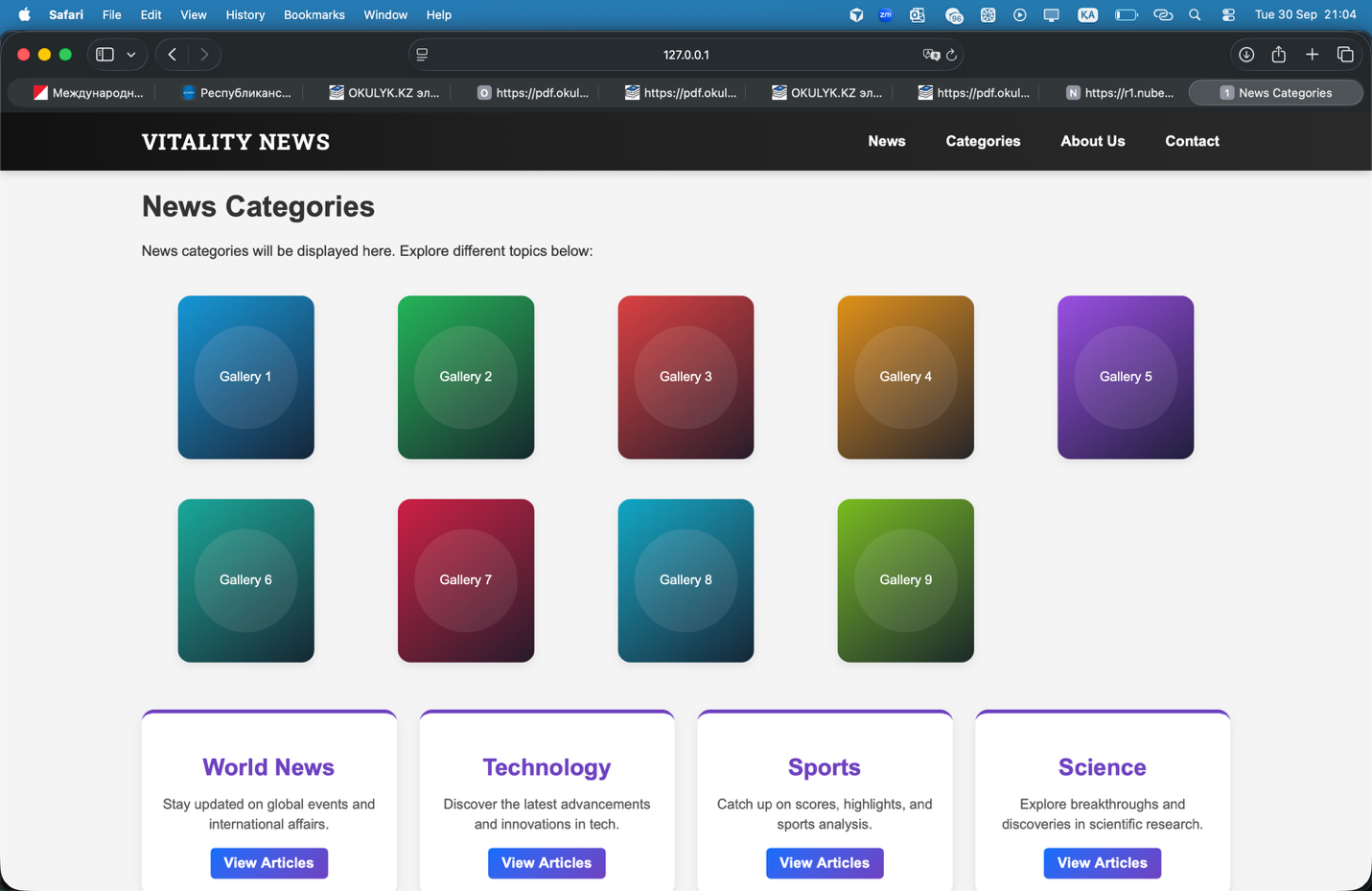
Key styles in styles/news.css: - .page-grid { display: grid; grid-template-areas: "header" "main" "footer"; } - .grid-header, .grid-main, .grid-footer assign areas.



### Task 4: Image Gallery (Categories)

* Built a 3+ column responsive image gallery using CSS Grid.
* Added hover overlay caption and smooth scale effect.

Key styles in styles/categories.css: - .gallery-grid { display: grid; grid-template-columns: repeat(auto-fill, minmax(200px, 1fr)); gap: 12px; } - .gallery-item img { object-fit: cover; } - .gallery-item figcaption overlay with gradient and transition.



## Responsiveness

* Flex containers wrap gracefully (.card-row).
* Grids use auto-fit/auto-fill and minmax for fluid columns.
* Media queries are minimal; layout adapts naturally.

## UX and Engagement

* Bright, accessible color palette via CSS variables in styles/style.css.
* Clear hover feedback for links, buttons, and cards.
* Consistent spacing and rounded corners.

## How Each Task Was Achieved (Step-by-Step)

1. Navigation: Converted header to Flexbox and set gaps and hover states.
2. Cards: Added semantic card markup and Flexbox; ensured equal heights with flexible body.
3. News Layout: Defined grid container with areas; placed header and article grid inside.
4. Gallery: Implemented grid with fixed-height images via object-fit: cover; overlay captions on hover.
5. Polishing: Unified palette using :root CSS variables, gradients, and shadows.

## Code Snippets

Header Flexbox (excerpt):

nav { display: flex; justify-content: space-between; align-items: center; }  
nav ul { display: flex; }  
nav ul li a { padding: 8px 12px; border-radius: 999px; }

Screenshot (insert below):

|  |
| --- |
| Header Flexbox |

Header Flexbox

Card Row (excerpt):

.card-row { display: flex; flex-wrap: wrap; gap: 20px; }  
.card { display: flex; flex-direction: column; }  
.card-body { flex: 1 1 auto; }

Screenshot (insert below):

|  |
| --- |
| Cards Flex |

Cards Flex

News Grid Areas (excerpt):

.page-grid { display: grid; grid-template-areas: "header" "main" "footer"; gap: 24px; }  
.grid-header { grid-area: header; }  
.grid-main { grid-area: main; }  
.grid-footer { grid-area: footer; }

Screenshot (insert below):

|  |
| --- |
| News Grid Areas |

News Grid Areas

Gallery Grid (excerpt):

.gallery-grid { display: grid; grid-template-columns: repeat(auto-fill, minmax(200px, 1fr)); gap: 12px; }  
.gallery-item img { width: 100%; height: 180px; object-fit: cover; }

Screenshot (insert below):

|  |
| --- |
| Gallery Grid |

Gallery Grid

## Final Reflection

This assignment solidified practical knowledge of Flexbox and Grid. We implemented a consistent, responsive layout system without frameworks, improved accessibility and visual clarity, and ensured maintainable styles via variables and reusable components.

## Appendix: Where to Find Changes

* Flexbox Nav and Cards: styles/style.css, index.html
* News Grid Areas: styles/news.css, html/news.html
* Categories Gallery: styles/categories.css, html/categories.html

## Screenshots

