

GAC011 Computing II: Web Development

GAC Computing

What this module is, and how it is marked

GAC011 is where you build and publish a real website. Six units take you from a design on paper through HTML, CSS and JavaScript to a launched site and a plan for promoting it.

Everything is assessed on **what you produce**: a **storyboard** 分镜脚本 and design, the site itself, and a **social media strategy** 社交媒体策略, plus coursework. There is no written exam.

- **Design counts as much as code.** The module's outcomes name aesthetics, layout, navigation and usability alongside the scripting.
- △ The site you design here is the thing GAC017 gives a back end and a database. A weak plan now costs time in the next module.

You can run every example in this module in the site's own code playground, and the HTML, CSS and JavaScript references there go deeper than these notes do.

Designing a website

- Start on paper. A **wireframe** 线框图 shows structure without colour, which is the only way to argue about structure.
- **Information architecture** 信息架构 is what goes where: the pages, and the path a visitor takes between them.
- **Visual hierarchy** 视觉层次 tells the eye what matters — through size, weight, spacing and contrast, not through decoration.
- Design for the **audience**, and say who they are before you choose anything.

Creating a website using HTML

- **HTML** 超文本标记语言 marks up meaning, not appearance. A heading is `<h1>` because it *is* a heading.
- An **element** 元素 has an opening tag, content and a closing tag; an **attribute** 属性 goes in the opening tag.
- **Semantic elements** 语义化元素 — `<header>`, `<nav>`, `<main>`, `<article>`, `<footer>` — tell a browser, a search engine and a screen reader what a region is.

- Every image needs **alt text** 替代文本. It is what a blind visitor hears and what a search engine reads.

Using CSS to enhance your website

- **CSS 层叠样式表** controls appearance, and keeping it out of the HTML is the whole point of the separation.
- A **selector** 选择器 chooses elements; a **declaration** 声明 sets a property to a value.
- The **box model** 盒模型 explains layout: content, then padding, then border, then margin.
- **Responsive design** 响应式设计 makes one page work on a phone and a laptop, usually with flexible widths and a **media query** 媒体查询.

Worked example. A card that is too wide on a phone.

The fix is not a fixed pixel width. Set **max-width** so the card never exceeds the screen, let it shrink below that, and add a media query only for the change that a smaller screen genuinely needs, such as stacking two columns into one.

Designing the flexible case first, and treating the media query as the exception, is what “responsive” means.

Adding interactivity with JavaScript

- **JavaScript** runs in the browser and changes the page after it has loaded.
- The **DOM** 文档对象模型 is the page as a tree of objects your code can read and change.
- An **event** 事件 — a click, a submit, a key press — triggers a **function** 函数 you wrote.
- Keep it small at this level: validate a form, toggle a menu, filter a list. A site that fails when JavaScript fails is a design error, not a coding one.

Launching your website

- **Hosting** 托管 puts your files on a server; a **domain name** 域名 points visitors at them.
- **Test before you launch:** every link, on a phone, on a browser that is not yours.
- **Accessibility** 无障碍 is not optional — contrast, alt text, keyboard navigation, and text that resizes.

- $\triangle$ Check what you are allowed to publish. An image or a font on your site needs a licence that permits it, exactly as a quotation in an essay needs a citation.

Social media marketing

- Decide the **audience** and the **objective** before choosing a platform. “Be on social media” is not an objective.
- A **content plan** 内容计划 says what you will post, how often, and who writes it.
- **Metrics** 指标 must connect to the objective: reach, engagement, and click-through to your site are different things, and only one of them is traffic.
- Measure, then change one thing. Changing everything at once teaches you nothing about which change worked.