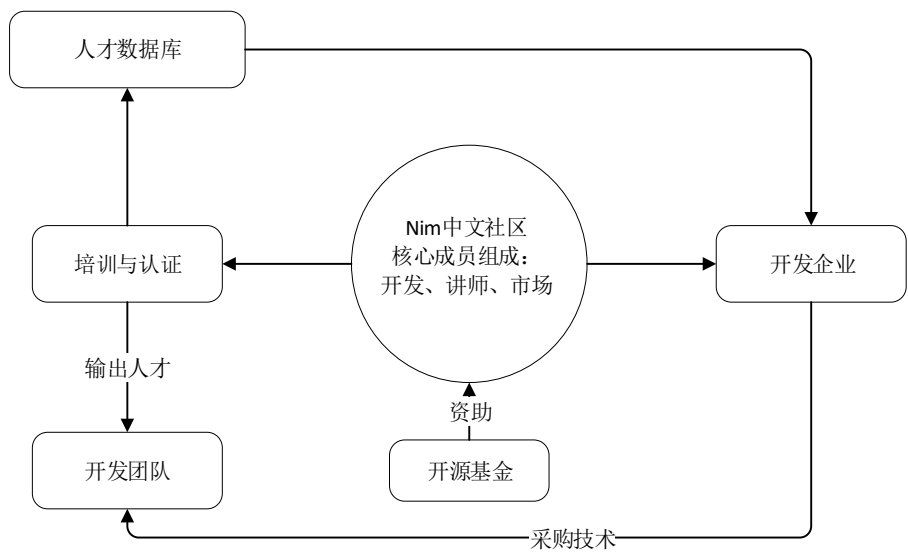


致 Nim 社区



以上是我们与开源基金达成的 Nim 能够运转起来的框架共识，从框架上看，社区核心人员贯穿了发展的前后期，然后是 Nim 认证。前期我们需要系统开发和 GUI 开发者，完成 Nim 认证系统的开发，并通过 Nim 认证吸引更多开发者。通过 Nim 认证的开发者进入开发团队，使用 Nim 生态库完成企业在 Web 前后端，系统开发、物联网等多领域应用的迁移和开发。

认证数据形成多技术指标索引的数据库，例如企业搜索掌握响应式 Web，会使用 websocket，css 变换与动画的开发人员，因为它们在做一个在桌面和移动端风格统一，具有细节动画过渡效果，有服务器推送消息的展示性网站。加上会使用 NPL 的条件，则可能是企业需要能够做自动生成网页版新闻的开发人员。

Nim 认证参考 AWS 认证，每周报名考试，认证内容以领域为单位，由 Andreas 和 Nim 委员会成员确认。预设 Web 前端，系统开发、嵌入式、移动开发、AI 五个方面，每个题目进行某个知识或技能上达到初学者(novice)、熟手(journeyman)、专家(expert)三个等级之一的测试，代表完成任务需要额外学习，能够以平均水平和时间完成任务，以较高水平和较短时间内完成任务并可能带来性能、算法、代码优化。

评估基于测评者领域和题目的完成度由自然语言处理 AI 自动生成。（前期由人工完成）

认证达到熟手程度的颁发签名证书，可选择进入 Nim 开发团队，赠送 Nim 体恤或口杯。

Nim 认证示例如下：

Certificate of Nim

Iron Man has passed the following expertise testing of Web Frontend domain with Certificate

of Nim, assessment attached below:

钢铁侠 同学经过 Nim 认证完成了 Web 前端开发的技能认证，结果如下：

HTML: Can make use of HTML tag and global attribution and events, can use precise tag to design Web page, and set proper tag attribution and events to interact with script and CSS. Know the difference and relationship between Web app and SSR.

HTML: 会使用 HTML 的标签和全局属性与事件，可以准确的使用标签设计 Web 页面，设置合适的标签属性和事件完成与脚本和 CSS 的交互，理解 Web 应用与 SSR 的区别与联系。

CSS: Can make use of CSS、CSS3 and custom CSS properties, can analyze layout and 2D animations of a Web page by looking. Require a certain period of time to learn and apply for 3D animation analysis and implement. Have a preliminary understanding of media queries, grid systems, flex layout, takes couples of weeks to learn and implement responsive css style.Unable to finish responsive website very soon, require couple of days to learn. Have a preliminary understanding of different browser CSS properties setting,can implement limited compatibility between browsers.

CSS: 会使用 CSS、CSS3 和自定义 CSS 属性，可以通过目测分析网站布局的 CSS 属性和平面动画，需要几天时间实现三维效果动画分析与实现。对媒体查询、栅格系统，flex 具有初步理解，需要几周时间学习完成响应式网站的 CSS 样式。对不同浏览器 CSS 属性设置有初步理解，可以实现有限的兼容性。

Javascript: Can make use of DOM operating and CSS property API, Can append,delete,replace element to a DOM tree and change the CSS properties. Can implement dynamic element Web page based on event dispatching model. Can use Web File API to read information, can use Ajax and FormData to upload file with progress bar.

Javascript: 会使用 Javascript 操作 DOM 和 CSS 属性的 API，可以正确的对页面 DOM 树中的元素进行动态追加、删除、替换、改变 DOM 元素的 CSS 属性，可以实现基于事件传播模型的动态元素 Web 页面，会使用 Web File API 读取文件信息、会使用 Ajax 与 FormData 实现带有进度条的文件上传。

Canvas API: Can make use of HTML Canvas API to draw line, circle, ellipse, arc , can use square and cubic bezier curve to draw arbitrary curve, can make translate, scale, rotate animation. Can implement linear gradient and radial gradient by looking.

Canvas API: 会使用 HTML5 Canvas API 绘制直线、圆形、椭圆、弧线、会使用二阶和三阶贝赛尔曲线绘制任意曲线、会制作移动，缩放、旋转动画。可以通过目测实现图形的线性渐变和径向渐变。

echarts: Can make use of (stacked) bar, pie, gauge and (stacked) line to express data. Can custom different theme of echarts. Can use specific echarts javascript file to reduce size.. Takes 0.5 hour the learn and apply a new type chart.

echarts: 会使用 echarts （堆叠）柱状图、饼状图、仪表盘、（堆叠）折线图表达数据。会定制不同的 echarts 颜色主题。会根据需要使用指定的 echarts javascript 文件以减少体积。

Websocket: Can make use of websocket non-persistent connection, simple study is required

to implement websocket persistent connection.

Websocket: 会使用 Websocket 短连接, 可以通过简单学习实现长连接和 Websocket 上传文件。

Routing: Can make use of Web History API to make arbitrary same source route, can implement page state pushing and popping, history record forward and backward.

路由: 会使用 Web History API 实现任意同源路由, 可以正确实现页面状态的弹入和弹出, 历史记录的前进、后退。

Macro: Can make elementary use of macro, couples of weeks are required to customize Karax or bugfix.

宏: 会初步使用宏, 需要几周时间学习对 Karax 进行定制或 bug 修复。

In conclusion, Iron Man can finish the Web frontend designing of a responsive website which has pushed data from server, has upload file function, simple animation, and maximum compatibility. Capability of using macro and underlying Web API makes him capable to create new Web frontend development tools to improve productivity.

综合, 钢铁侠同学可以在平均时间内实现从服务器推送数据, 具有文件上传功能、简单动画, 有限兼容性的响应式 Web 前端设计。

Certificate of Nim

Iron Man has passed the following expertise testing of System Programming domain with Certificate of Nim, assessment attached below:

钢铁侠 同学经过 Nim 认证完成了系统开发的技能认证, 结果如下:

File programming: Can make use of Linux file APIs, Doesn't have the knowledge of Windows system C++ file API.

文件编程: 会使用 Linux 文件 API。不了解 Windows 系统 C++ 文件 API。

Network programming: Can make use of Http/TCP/UDP/Websocket to communicate between client and server. Doesn't have the knowledge of corresponding Windows system C++ API.

网络编程: 会使用 Http、TCP、UDP, Websocket 通信, 可以实现客户端和服务端通信。不了解相应的 Windows 系统 C++ API。

Asynchronous programming: Can make user of POSIX select, epoll IO multiplexing, understand level trigger and edge trigger, can implement async communication based on POSIX. Doesn't have the knowledge of Windows C++ IO multiplexing API.

异步编程: 会使用 POSIX select, epoll IO 复用, 了解边缘触发和水平触发, 可以实现基于

POSIX 的系统异步通信，不了解 Windows 系统 C++ IO 复用 API。

Multi-threads programming: Can make use of POSIX thread, can implement multi-threads system. Doesn't have the knowledge of Windows C++ threads API.

多线程编程：会使用 POSIX 线程，可以实现多线程系统。不了解 Windows 系统 C++ 线程 API。

Inter-Process Communication: Can make use of Linux C/C++ IPC, doesn't have the knowledge of Windows C++ IPC API.

进程间通信 (IPC)：会使用 Linux C/C++ 系统进程间通信，不了解 Windows 系统 C++ 进程间通信 API。

Signal: Understand the functionality of signal and default behavior to pick precise signal to send.

信号：理解信号的作用和每种信号的默认行以选择准确的信号用来发送。

Garbage Collection(GC): Master memory allocating of stack and heap, understand the cause of garbage producing, GC algorithm and effect.

垃圾回收 (GC)：掌握堆栈的内存分配机制，了解垃圾产生的原因，垃圾回收算法以及产生的影响。

Database: Can make use of CRUD, understand index, know InnoDB and MyISAM, understand read and write lock, master transaction, can use inner, left, right and foreign key, know the rules of Mysql performance optimization, can make use of JSON field type, understand MySQL cluster, reading and writing isolation, master-slave synchronization.

数据库：会使用 Mysql CRUD，理解索引，了解 InnoDB 和 MyISAM 表，理解读写锁，掌握事务，会使用内、左右连接和外键，了解 MySQL 性能优化规则，会使用 MySQL JSON 字段，理解 MySQL 集群，读写分离，主从同步。

Principle of Compiler: Has a preliminary understanding of compiler principle.

编译原理：对编译原理有初步理解

Operating System: Master operating system kernel, memory management.

操作系统：掌握操作系统内核、内存管理。

Macro: Master Nim AST manipulation.

宏：掌握 Nim AST 操作。

Finite-state machine(FSM), his language will be expressive and productive.)

综合，钢铁侠同学可以完成系统开发中的网络编程与数据库设计，可以使用宏创建特定领域语言 (Domain-Specific-Language)，解决跨领域的问题。

Nim t-shirt 和口杯

