

Overview on Composable UI

نگاهی کلی به رابط کاربری ساختنی

حسین خادمیان^۱

^۱ دانشجوی کارشناسی، مهندسی کامپیوتر، دانشگاه شیراز، شیراز، me@hkhademia.ir

۱۳۹۹/۰۴/۱۸

چکیده

در این مقاله، به معرفی روابط کاربری Composable، مزایا و معایب این سیستم، علت چرخش شرکت های بزرگ فناوری به استفاده از این شیوه ساخت روابط کاربری و همچنین نگاه و طریقه آنها در پیاده سازی می پردازیم.

کلمات کلیدی

UI, Composable UI, Flutter, Jetpack Compose, Swift UI, React JS, React Native.

۱ مقدمه

شیوه کنونی تولید روابط کاربری بر مبنای تعریف عناصر صفحه صفحه نمایش به صورت اشیایی شناخته شده و با جداسازی widget های گوناگون مانند Button، ListView، DropBox، CheckBox، ... استوار است.

این شیوه توصیف لایه های کاربری در پلتفرم های گوناگون، جهت پیاده سازی رابط کاربری برنامه ها در سیستم عامل ها دستگاه های مختلف (از جمله کامپیوتر شخصی، تبلت، موبایل، صفحات وب و ...) پیاده سازی می شد.

۲ تکنولوژی های پیشین

جهت پرداختن به مزایا و معایب لایه های کاربری ساختنی و همچنین بررسی تفاوت های آن با شیوه های پیاده سازی کنونی ابتدا باید آشنایی کلی با وضعیت فعلی داشته باشیم. در ادامه به معرفی چند مورد از شیوه های پر استفاده که با آنها کار کرده ام جهت بررسی میپردازم:

۱-۲ HTML در صفحات وب

صفحات وب مجموعه ای از المنت هاست^۱ که هرکدام مشخص کننده قسمتی از ویژگی های صفحه درحال نمایش هستند. صفحات وب از چندین تگ کلی:

۱. `<HTML>`: این تگ در برگیرنده کل اطلاعات صفحه است

۲. `<BODY>`: این تگ در برگیرنده کل محتوای قابل نمایش صفحه است

```
1 <html id="rtl">
2 <head>
3 <title>HTML layout using Table (deprecated)</title>
4 <link rel="stylesheet" href="styles.css">
5 <style>
6 </style>
7 </head>
8 <body>
9 <div id="page-wrap">
10 <div id="header" colspan="2">Header</div>
11 <div id="main">
12 <div id="sidebar" rowspan="2">Sidebar</div>
13 <div id="content">
14 <div id="sub-nav">sub menu</div>
15 <div id="article">
16 <form id="login" method="post">
17 Username: <input type="text" name="user" value="" />
18 Password: <input type="password" name="pass" value="" />
19 <input type="submit" name="login" value="Login"/>
20 </form>
21 </div>
22 <div style="clear: both;"></div>
23 </div>
24 <div id="footer" colspan="2">Footer</div>
25 </div>
26 </body>
27 </html>
```

شکل ۱: نمونه کد HTML

۳. `<HEAD>`: در این بخش اطلاعات صفحه مانند عنوان و استایل صفحه معرفی می شوند

از خواص این نوع تعریف لایه کاربری: تعریف قسمت های مختلف صفحه (دکمه ها، فرم ها، زیرصفحه و ...) در کنار مشخصات و ویژگی های آنها (رنگ، فونت، ...) و حتی کد های رویداد های مختلف در کنار هم است.

```

<ResourceDictionary
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    xmlns:wpfToolkit="http://schemas.microsoft.com/wpf/2008/toolkit"
    xmlns:Primitives="clr-namespace:Microsoft.Windows.Controls.Primitives;assembly=WPFToolkit">

    <ResourceDictionary.MergedDictionaries>
        <ResourceDictionary Source="Brushes.xaml"/>
    </ResourceDictionary.MergedDictionaries>

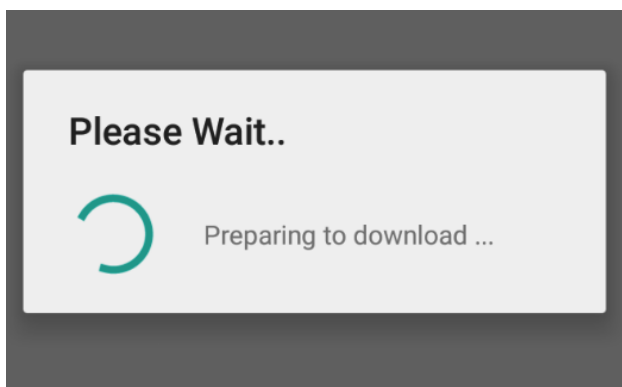
    <!-- Value converters -->
    <BooleanToVisibilityConverter x:Key="bool2VisibilityConverter" />

    <!-- DataGridColumnHeader Right Gripper Style -->
    <Style x:Key="GridColumnHeaderRightGripperStyle" TargetType="{x:Type Thumb}">
        <Setter Property="Width" Value="8"/>
        <Setter Property="Background" Value="Transparent"/>
        <Setter Property="Cursor" Value="SizeWE"/>
        <Setter Property="Template">
            <Setter.Value>
                <ControlTemplate TargetType="{x:Type Thumb}">
                    <Border Padding="{TemplateBinding Padding}"
                        Background="{TemplateBinding Background}" />
                </ControlTemplate>
            </Setter.Value>
        </Setter>
    </Style>

    <!-- DataGridColumnHeader Left Gripper Style -->
    <Style x:Key="GridColumnHeaderLeftGripperStyle"
        BasedOn="{StaticResource ColumnHeaderRightGripperStyle}"
        TargetType="{x:Type Thumb}">
        <Setter Property="Width" Value="8"/>
        <Setter Property="Background" Value="Transparent"/>
    </Style>

```

شکل ۳: نمونه کد XAML



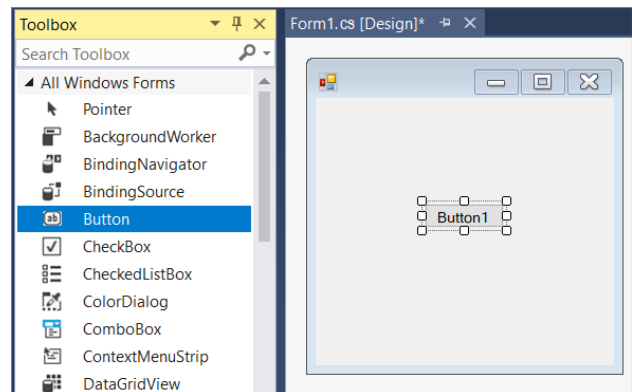
شکل ۴: نمونه دیالوگ در حال بارگذاری قبل از چرخش

نمایش داده شده به کاربر به طور مستقیم ندارد. بلکه با استفاده از ویجت های ارایه شده هر پلتفرم و با اتکا به رویداد های گزارش شده آنها به بروز رسانی وضعیت مد نظر می پردازد.

این شیوه از مدیریت وضعیت و اتکا به پلتفرم های گوناگون (با امکان آپدیت و تغییرات) باعث مشکلات بسیاری در برنامه ها می شود.

اکثر ما تا کنون با برنامه هایی که برخلاف انتظار برنامه نویس به وضعیتی نامطلوب رفته اند (فرضا یک دیالوگ "در حال بارگزاری" در یک برنامه اندرویدی که با چرخش صفحه ناپدید می شود اما از دید برنامه هنوز باید نمایش داده می شد). یکی از مهمترین دلایل این تجارب، نه از عدم آگاهی توسعه دهنده بلکه از **غیر قابل پیش بینی بودن پلتفرم** ایجاد می شود.

جهت مدیریت اینگونه وضعیت های از قبل پیشبینی نشده که اکثرا به علت نقص در پلتفرم، بروز نبودن پلتفرم در دستگاه کاربر (مثل نسخه اندرویدی دستگاه)، عدم شناخت کافی توسعه دهنده از ویژگی ها و عکس العمل های هر کامپوننت در شرایط گوناگون نشات می گیرد، مستلزم به بکارگیری تیم آزمایش محصول و همچنین فرایند پیچیده تر دریافت گزارش های خطا از کاربر می شود. حتی گاهی مدیریت یک وضعیت ناخواسته از خود منطق برنامه پیچیدگی بیشتری پیدا میکند. یکی دیگر از موارد مهم، **Reusability** کد ها و کامپوننت هایی که یک برنامه



شکل ۲: نمونه طراحی با ویرایشگر WinForm

مزایا:

۱. این مدل توصیفی است، یعنی برنامه نویسی بدون نیاز به کد نویسی به توصیف لایه کاربری می پردازد.
۲. زبان ساده و خوانا، قابل استفاده برای نا-برنامه نویسان (طراحان، دانش آموزان و ...)
۳. چشم پوشی از خطا: اگر در توصیف لایه کاربری خطایی (در بارگزاری یا حتی سمت برنامه نویس) رخ دهد کل برنامه بی استفاده نمی شود.

معایب:

۱. فقط امکان تولید صفحات استاتیک به تنهایی دارد
۲. مخلوطی از تمامی منطق برنامه، ظاهر کاربری، کنترل و اطلاعات. باعث عدم امکان توسعه خطی در پروژه های بزرگ و سازمانی می شود.
۳. نیازمند دیگر ابزار جهت پویا سازی مثل جاوا اسکریپت، activeX و ...
۴. صاحب و مدیر StateMachine است و برنامه نویس فقط با event ها از تغییرات وضعیت با خبر می شود. و کنترل کاملی روی آن ندارد. این مسئله باعث به وقوع پیوستن States nonDeterministic می شود.

۲-۲ WindowsForms, UWP, WPF

پلتفرم WindowsForms تکنولوژی قدیمی بر بستر WindowsAPI ارایه می شود و به طور کلی ابزاری جهت پیاده سازی رابط کاربری گرافیکی برای برنامه های ویندوزی (GUI) است. در این پلتفرم جهت توصیف لایه کاربری مستقیما با LayoutEditor نسبت به تعریف رابط کاربری اقدام می کنیم. ایده پشت این پلتفرم Form-Based بودن است. به این شیوه که تمامی صفحاتی که کاربر با آن تعامل می کند در حقیقت فرمی جهت ورود اطلاعات و دریافت بازخورد از اوست. نسخه بروزتر از این پلتفرم UWP و WPF که به ترتیب برای توسعه برنامه های چند سکویی و دسکتاپ طراحی شده اند دارای زبانی جهت توصیف لایه کاربری به نام XAML هستند. این شیوه شباهت بسیاری به صفحات HTML داشته و بیشتر مزایا و معایب آن در این شیوه هم صادق است.

۳ زمینه

تا کنون با تعدادی از شیوه های تولید رابط کاربری قدیمی آشنا شدیم. در اکثر این شیوه ها برنامه نویس هیچ گونه مدیریتی بر وضعیت ماشین (برنامه) و محتوای

نویس تولید و استفاده می کند است. در شیوه سنتی به دلیل وابستگی کامل برنامه نویس به زمینه ای که در آن یک کامپوننت نوشته می شود (از جمله موقعیت قرار گیری در صفحه، المنت های دیگر صفحه و ...) امکان استفاده از یک کامپوننت در دیگر جاهای برنامه کمتر و فرایند آماده سازی یک کامپوننت برای استفاده در پروژه های دیگر خیلی هزینه زیادی دارد.

اکثر مشکلات اشاره شده در این قسمت و قسمت های قبل از عدم کنترل برنامه نویس بر وضعیت کنونی برنامه و همچنین متمرکز نبودن مرجعیت درستی وضعیت در کل برنامه است. برای مثال: به این صورت که از دید برنامه نویس چک باکس شماره ۱ فعال و چک باکس ۲ غیر فعال است. با تغییر این انتخاب توسط کاربر در اکثر پلتفرم ها ابتدا وضعیت تغییر پیدا کرده و سپس تغییر جدید به برنامه نویس از طریق Event ها اطلاع داده می شود. و برنامه نویس کنترلی روی این تغییر ندارد.

۴ راه کار

راه حل این مسئله فرستادن کل فرایند رسیدگی و مدیریت وضعیت ها به برنامه نویس و رندرینگ و ... به پلتفرم است. همچنین کامپوننت های تشکیل دهنده لایه کاربری هیچ گونه وضعیتی آشکار (قابل مدیریت توسط برنامه نویس) نگه داری نکنند. به این ترتیب حتی تغییر وضعیت یک دکمه رادیویی، تایپ در یک تکست باکس هم با اطلاع و موافقت برنامه نویس صورت میگیرد.

برای پیاده سازی این شیوه تولید کامپوننت های لایه کاربری نیاز است که تغییری در نحوه نگاه به لایه های کاربری به طور کلی بیاندازیم. از جمله:

۱. هر کامپوننت فقط باید به نقش انحصاری خود بپردازد و در کار بقیه قسمت ها تغییری ایجاد نکند.

۲. اطلاعات مورد نیاز جهت پیاده سازی به کامپوننت به آن داده می شود. همچنین کامپوننت از هرگونه تغییری در اطلاعات باید آگاه شود.

۳. کامپوننت ممکن است جهت اجرای نقش مشخص شده خود، دارای وضعیتی داخلی باشد که البته هیچ تاثیری در وضعیت کلی برنامه نباید بگذارد.

۴. کامپوننت ها نباید وابستگی به والد یا فرزندان خود داشته باشند و به تنهایی و در هر محیط و usecase بتوانند وظیفه مربوطه را انجام دهند.

یکی از دیدگاه های مفید در این زمینه به این صورت مطرح میشود که:

UI is a function of State

به این معنی که هرگونه تغییر در وضعیت برنامه در ظاهر کاربری اثر میگذارد و نه برعکس. ظاهر کاربری هیچگونه مسیولیتی در زمینه صحت سنجی، بروز رسانی، مدیریت، بارگذاری و پردازش اطلاعات ورودی و خروجی و همچنین وضعیت فعلی برنامه ندارد و تنها مسئولیت آن به نمایش گذاشتن وضعیت فعلی اطلاعات برنامه است.

در ادامه چندین پیاده سازی این راه کار را بررسی می کنیم:

۱-۴ ویژگی متن اصلی مقاله

متن اصلی مقاله خود می تواند در چهار بخش مختلف دسته بندی شود.

۱-۱-۴ ویژگی های بخش پیش نیازها

در صورتی که نویسندگان لازم است که یک مطلب را برای خوانندگان به عنوان پیش نیاز و پیش زمینه فهم روش پیشنهادی خود ارایه کنند این موارد را در این بخش می توانند بیاورند. به عنوان مثال اگر شما می خواهید یک روش مبتنی بر یادگیری SVM در حوزه نهان کاوی تصویر ارایه دهید، می توانید توضیحات مقدماتی در مورد SVM را در این بخش بیان کنید. البته توصیه کلی بر این است که در حد امکان از آوردن مطالبی که خواننده می تواند آن را براحتی با خواندن مراجع دیگر بدست آورد، پرهیز کنید.

۲-۱-۴ ویژگی های بخش مدل سامانه و فرضیات

در این بخش نویسندگان باید مدل سامانه را به صورت دقیق مشخص کنند. اگر سامانه مورد بررسی و یا کار آن ها دارای فرضیات مشخصی است باید آن را در این قسمت مشخص کنند. شما باید سامانه را به گونه ای مدل کنید و فرضیات را به نحوی تعیین کنید که چندان به دور از ذهن و بدور از مدل ها و فرضیات مقالات گذشته نباشد.

۳-۱-۴ ویژگی بخش روش پیشنهادی

در این بخش شما باید به دقت و گام به گام به معرفی کار انجام شده و نوآوری صورت گرفته در مقاله مبادرت بورزید.

برای وارد کردن الگوریتم و سودو کد از محیط algorithm استفاده کنید. توصیه می شود که نویسندگان حتما سعی کنند روش خود را به صورت شماتیک با یک فلوجارت و یا استفاده از محیط الگوریتم نمایش دهند. نمونه ای از فلوجارت نیز در شکل ۵ نشان داده شده است. دقت کنید که در وارد کردن هرگونه تصویری در مقاله از قرار دادن option هایی به مانند H، ht و ... خودداری کنید. نویسندگان دقت داشته باشند که می بایست روش پیشنهادی خود را به صورت ساده، واضح و مشخص بیان کنند. سعی کنید در این بخش روش پیشنهادی را در حالت کلی مورد بررسی قرار دهید، و سپس در بخش های بعدی به جوانب آن بپردازید. اگر روش پیشنهادی شما دارای چندین مرحله (فاز) است، بهتر است هر مرحله را در یک زیربخش به صورت مجزا مورد بررسی قرار دهید.

۴-۱-۴ ویژگی های بخش تحلیل و ارزیابی

در صورتی که روش پیشنهادی حسی نبوده و مبتنی بر استدلال ریاضیاتی باشد، می توان به ارزیابی عملکرد آن در سامانه و نحوه بهبود نتایج نسبت به کارهای گذشته به طور کمی و کیفی پرداخت. بدیهی است اگر مدل پیشنهادی حسی بوده و قابل استناد ریاضیاتی نباشد، تنها می توان به صورت کیفی به ارزیابی عملکرد پرداخت.

۲-۴ ویژگی های بخش شبیه سازی

از آنجایی که بسیاری از تحقیق ها به منظور حل مساله ای عملی پی گیری می شوند، این بخش از اهمیت خاصی برخوردار است. در این بخش به معرفی شبیه سازی صورت گرفته و ارائه نتایج به صورت مطلوب (جدول، نمودار و ...) پرداخته می شود. در مواردی که طرح پیشنهادی اثبات شده است، نتایج باید با تقریب خوبی با مدعا یکسان باشد. در مواردی که طرح پیشنهادی اثبات نشده و به طور حسی و مبتنی بر برخی پیش فرض ها مطرح شده است، اهمیت این بخش بیشتر از حالت قبل است، چرا که نتایج شبیه سازی تنها مستند نویسنده و به نوعی موید مدعای ثابت نشده

الگوریتم ۱ الگوریتم ثبت تصویر لوکاس-کاناد مبتنی بر بهینه‌سازی گوس-نیوتون (LK-GN).

Input: The reference image I and template image T .

Output: Registration parameters $\mathbf{p} = (p_1, \dots, p_n)^T$ as the warp model W .

```

1: repeat
2:   Warp  $I$  with  $W$  to compute  $IW$ .
3:   Compute the error image  $T(x) - IW$ 
4:   Warp the gradient  $\nabla I$  with  $W$ .
5:   Evaluate the Jacobian  $Wp$  at  $(\mathbf{x}; \mathbf{p})$ .
6:   Compute the steepest descent images  $\nabla IWp$ .
7:   loop
8:     <text>
9:   end loop
10:  Compute the Hessian matrix using (2).
11:  for <condition> to <condition> do
12:    <text>
13:  end for
14:  while <condition> do
15:    <text>
16:  end while
17:  Compute  $\Delta \mathbf{p}$  using (3)
18:  return Update the parameters  $\mathbf{p} \leftarrow \mathbf{p} + \Delta \mathbf{p}$ 
19: until  $\|\Delta \mathbf{p}\| \leq \epsilon$  or Reaching to Maximum Iteration allowed

```



شکل ۵: چارچوب کلی روش پیشنهادی.



شکل ۶: زیرنویس نمودارها باید کامل و جامع باشد، به عبارت دیگر خواننده بتواند با دیدن نمودار و خواندن زیرنویس آن پی به تمامی اطلاعات نهفته در نمودار ببرد.

وی است. این حالت اخیر از دهه آخر قرن بیستم تا کنون به وفور در مقالات معتبر پی‌گیری شده است تا جایی که در صورت تایید نتایج با شبیه‌سازی، آن را به نوعی اثبات برای طرح پیشنهادی در نظر می‌گیرند.

برای آوردن اشکال در کنار یکدیگر می‌توانید از محیط subfigure استفاده کنید. در شبیه‌سازی‌ها می‌بایست نویسندگان به صورت دقیق و صریح پیکربندی شبیه‌سازی و مجموعه داده‌ای که مورد استفاده قرار داده‌اند را با ذکر منابع و مراجع مورد نیاز، ذکر کنند. در ضمن پارامترهای مورد استفاده برای شبیه‌سازی باید در همین بخش شبیه‌سازی تعریف شود. به عنوان مثال اگر شما از پارامتر MSE (mean squared error) استفاده می‌کنید باید آن را در این بخش تعریف کنید. شکل‌های شبیه‌سازی باید واضح و مشخص باشد. دقت کنید به دلیل این که در نهایت مقالات پذیرفته شده به صورت سیاه و سفید چاپ خواهد شد، بدین‌سان از تمایزگذاری با رنگ‌های مختلف بین نمودارها کافی نخواهد بود. یک نمونه از تمایزگذاری مناسب را می‌توانید در نمودار ۶ و ۷ ملاحظه کنید.

تمامی نمودارهای قسمت شبیه‌سازی باید دارای Legend باشند. محورهای نمودارها همگی باید دارای برجسب مناسب و همچنین شماره‌گذاری مناسب باشد. در \LaTeX سعی کنید نمودارهای خود را با کیفیت pdf وارد کنید و از قراردادن نمودارهای با کیفیت jpg و یا کیفیت تصویری خودداری کنید. بسیاری از ابزارهای شبیه‌سازی به شما خروجی pdf را می‌دهند.

استفاده از نمودارهای نوین به مانند Box-Plot به شدت مورد استقبال قرار می‌گیرد. نکته مهم در شبیه‌سازی انجام پذیرفته در مقاله این است که نویسندگان



(ج)

(ب)

(آ)

شکل ۷: (آ) این شکل در این مورد سخن می گوید که، (ب) در مورد این شکل اکنون صحبت کنید....، (ج) این شکل برای

این جمله دقت کنید: «آهنگی که شما از فروشگاه iTunes دریافت می کنید توسط قالب DRM اپل که يك قالب فایل AAC انحصاری و محافظت شده است که اپل مجوز استفاده از آن را به هیچ کس نمی دهد، محافظت می شود». این جمله در واقع از سبک نگارش زبان انگلیسی پیروی می کند و به هیچ وجه برای جملات پارسی مناسب نیست. به راحتی می توان این جمله را به این صورت بازنویسی کرد: «آهنگی که شما از فروشگاه iTunes دریافت می کنید توسط قالب DRM اپل محافظت می شود. این قالب يك قالب فایل AAC انحصاری و محافظت شده است، و اپل مجوز استفاده از آن را به هیچ کس نمی دهد».

جداسازی اجزای مختلف يك جمله نیز نقش زیادی در فهم آسان آن دارد. ویرگول میتواند اجزای يك جمله را در جایی که نیاز به مکث هست، ازهم جدا کند؛ حال آن که نقطه ویرگول برای جداسازی دوجمله که با هم ارتباط معنایی دارند، بکار میرود. نقطه نیز برای جدا کردن جملات مورد استفاده قرار میگیرد. درکاربرد هالالین (پراتز) باید توجه شود که عبارت داخل آن برای توضیحی است که از اجزای جمله محسوب نشده و درصورت حذف خللی به آن وارد نمیشود. درمقابل، گیومه برای برجسته کردن جزئی از جمله بکار میرود.

تا جای ممکن از بکار بردن کلماتی مثل «میباشد»، «گردید»، و «بوده باشد» پرهیز شود. به جای آن ها اغلب می توان از کلمات ساده و روان مثل «است» و «شد» استفاده کرد. بکارگیری کلمات دشوار و غیرمعمول تنها باعث پیچیده شدن جمله و دشوار شدن فهم آن می شود.

برای کلمات فنی تا حد امکان از معادل های پارسی استفاده شود. بدون تردید کلمه «پردازش» زیاتر از «پروسس» است، و یا کلمه «ریزپردازنده» از «میکروپروسسور» مناسب تر است. در چنین مواقعی اگر احتمال می دهید خواننده با معادل پارسی آشنا نیست، از پانویس برای نوشتن معادل انگلیسی استفاده کنید. این کار را در اولین کاربرد معادل های پارسی انجام دهید. مثل کره راهنما^۴

تا حد امکان از کلمات انگلیسی در جملات استفاده نکنید. مثلاً بجای نوشتن Microsoft میتوانید بنویسید: «میکروسافت». اگر ناچار شدید در يك جمله از کلمات انگلیسی استفاده کنید، حتماً فاصله کافی بین آن ها و کلمات پارسی را رعایت کنید.

تنها به گزارش میانگین خطا بسنده کرده اند. از دیدگاه علم شبیه سازی این کار نمی تواند اطمینان خواننده را به شبیه سازی جلب کند. به عنوان مثال دو کلاس را در نظر بگیرید که نمرات سه دانشجوی آن ۹، ۱۰ و ۱۱ است و کلاس دیگر ۵، ۱۰ و ۱۵. هر دو کلاس میانگین یکسانی دارند، اما واقعا بین نمرات این دو کلاس تفاوت زیادی وجود دارد. به عنوان پیشنهاد می توانید از نمودار Box-Plot استفاده کنید و یا حداقل بازه اطمینان نتایج را گزارش کنید. نمونه ای از یک Box-Plot در شکل ۷ نشان داده شده است.

سعی کنید از قرار دادن کدهای شبیه سازی چه در قسمت شبیه سازی و چه در قسمت روش پیشنهادی به شدت خودداری کنید. نویسندگان به صورت اختیاری می توانند کدهای شبیه سازی خود را در یک وبسایت قرار داده و به آن در مقاله پیوند دهند.

۳-۴ ویژگی های بخش نتیجه گیری

در بخش نتیجه، نکات مهم انجام شده در کار بصورت خلاصه مرور و نتایج به دست آمده توضیح داده شوند. همچنین در این بخش باید سهم علمی مقاله بصورت واضح بیان شود. هرگز عین مطالب چکیده را در این بخش تکرار نکنید. نتیجه می تواند به کاربردهای پژوهش انجام شده اشاره کند؛ نکات مبهم و قابل پژوهش جدید را مطرح کند؛ ویا گسترش موضوع بحث را به زمینه های دیگر پیشنهاد دهد.

۴-۴ ویژگی بخش پیوست ها

بخش پیوست ها يك بخش اختیاری است و شماره گذاری نمی شود. موضوع های مرتبط با مقاله که در یکی از گروه های زیر قرار گیرند، می توانند در بخش ضمایم آورده شوند.

- اثبات ریاضی فرمول ها یا الگوریتم ها.
- داده ها و اطلاعات مربوط به مطالعه موردی.
- نتایج کار دیگر محققان و داده های مربوط به مقایسه آن ها.
- سایر موضوع های مرتبط که جزء بخش های اصلی مقاله نباشند.

۵ قواعد نگارشی

شیوایی و رسایی نوشتار در گرو ساده نویسی است. تلاش شود در متن مقاله از جملات رسا، گویا، و کوتاه استفاده شود و از نوشتن جملات تودرتو پرهیز شود. به

۱-۵ علامت‌گذاری

اگر ρ بیانگر چگالی تخمینی باشد، خواهیم داشت:

$$\rho = \frac{m}{A}, \quad (۱)$$

که در آن m جرم تخمینی و A سطح آن است.

اگر در مقاله شما نمادهای ریاضی متنوعی مورد استفاده قرار می‌گیرد، حتماً سعی کنید جدول نمادها در متن خود داشته باشید. مثل جدول ۱.

جدول ۱: نمادهای مورد استفاده در مقاله

نماد	توضیحات
N	تعداد کل گره‌های شبکه
\mathbb{R}^{++}	مجموعه اعداد حقیقی مثبت بزرگتر از صفر
ρ_t	چگالی شبکه در لحظه t
$\Pr[A]$	احتمال رخداد رویداد A

۲-۵ املا

درستی نوشتار بر پایه املاي زبان پارسی ضروری است. در این بخش برخی از موارد اشتباه متداول را یادآوری می‌کنیم. می‌توانید اطلاعات دقیق‌تر را با مراجعه به کتابهای نوشته شده در این زمینه پیدا کنید.

در افعال حال و گذشته استمراری باید دقت شود که «می» از جزء بعدی فعل جدا نماند. برای این منظور می‌توانید از نیم‌فاصله استفاده کنید.

در مورد «ها» ی جمع نیز دقت کنید که از کلمه جمع بسته شده جدا نوشته شود؛ مگر در کلمات تک هجایی مثل «آنها». برای جدانویسی نیز از فاصله متصل استفاده کنید. مثلاً «پردازنده‌ها» را بصورت «پردازنده‌ها» بنویسید.

جمع بستن کلمات پارسی یا لاتین با قواعد زبان عربی اشتباه است. بنابراین «پیشنهادهات» و «اساتید» اشتباه و درست آنها «پیشنهاده‌ها» و «استادان» است.

بهتر است همواره حرف اضافه «به» از کلمه بعدی خود جدا نوشته شود، مگر آن که این حرف جزء یک فعل یا صفت یا قید باشد؛ مانند: «بکار بستن»، «بجا» و «بندرت».

در مورد کلمات حاوی همزه قواعدی وجود دارد که پرداختن به آنها در این مقاله نمیگنجد، اما برای نمونه به املاي کلمات «مسأله»، «منشأ» و «رئیس» دقت کنید. همچنین، همزه در انتهای کلماتی که به الف ختم میشوند، نوشته نمیشود و در صورت اضافه شدن به کلمه بعدی، از «ی» استفاده میشود: «اجرا شده»، و «اجرای برنامه».

۳-۵ شکل‌ها و جدولها

شکل‌ها و جدولها باید دارای عنوان باشند. عنوان شکل‌ها در زیر شکل و عنوان جدولها در بالای جدول قرار می‌گیرند. در صورتی که از شکل‌ها یا جدولهای سایر منابع استفاده می‌کنید، باید حتماً شماره آن مرجع را در عنوان شکل یا جدول ذکر کنید.

در هنگام ارجاع به شکل یا جدول از شماره آن استفاده کنید و از بکار بردن عباراتی همچون «شکل زیر» پرهیز کنید. تمام جدولها و شکل‌ها باید در متن مورد ارجاع قرار گیرند.

۴-۵ فرمول‌ها و عبارات ریاضی

برای هر فرمول باید يك شماره در نظر گرفته شود. این شماره را در داخل يك جفت هالالین و بصورت راست‌چین قرار دهید. در \LaTeX شماره‌گذاری به صورت خودکار انجام می‌پذیرد. تمام متغیرها، پارامترها، و نمادهای يك عبارت ریاضی باید توضیح داده شوند. اگر قبل از نوشتن فرمول این کار انجام نشده است، باید بلافاصله پس از فرمول این توضیحات بیان شوند. مثال زیر را در نظر بگیرید.

۶ نکاتی در مورد نوشتن مقاله با \LaTeX

نویسندگانی که با محیط \LaTeX و نحوه کار با آن آشنایی ندارند می‌توانند به سایت www.parsilatex.com مراجعه کنند. مطالب مفید بسیاری در این زمینه در سایت مذکور قرار داده شده است. همچنین تمامی سوالات و اشکالات مرتبط با این استایل و در حالت کلی مرتبط با \LaTeX را می‌توانید در تالار پرسش و پاسخ همین سایت مطرح نمایید.

برای کار با این استایل در ابتدا چندین گام را می‌بایست بردارید:

۱. نصب یک کامپایلر به مانند TeXLive. دقت کنید این استایل را می‌توانید توسط TeXLive 2014 ویا TeXLive 2015 بروز شده استفاده نمایید. برای دانلود TeXLive می‌توانید به سایت TeXLive مراجعه کنید. در ضمن قابلیت خرید اینترنتی و ارسال پستی این مجموعه از طریق سایت parsilatex.com نیز فراهم شده است، بدین منظور به آدرس زیر مراجعه کنید.

بخش خرید سایت پارسی‌لاتک

۲. نصب یک ویرایشگر مناسب برای زبان فارسی به مانند TeXStudio.

نصب فونت‌های IRMitra و Times New Romans.

۳. کامپایل فایل نمونه ارائه شده. نویسندگان در صورتی سه گام قبلی را با موفقیت پشت‌سر گذاشتند که بتوانند فایل نمونه قرار داده شده را با موفقیت و به صورت Normally کامپایل کنند.

لازم به ذکر است که تمامی عبارت‌ها و کلمات انگلیسی در فایل \LaTeX باید درون دستور $\text{\texttt{Ir}}$ قرار گیرند. مثلاً: Support vector و Machine Learning. دقت کنید که تمامی کلمات و عبارت‌های اختصاری می‌بایست در اولین فراخوانی به صورت باز شده ارائه شود. به عنوان مثال در اولین مکانی که شما از کلمه اختصاری SVM استفاده می‌کنید باید آن را به صورت SVM (support vec- tor machine) بنویسید. به صورت معمول تمامی حروف حالت باز شده باید با حرف کوچک نوشته شود.

ترجیحا توصیه می‌شود که نویسندگان از قراردادن پاورقی در مقاله پرهیز کنند، اما در صورت نیاز شما می‌توانید با دستور footnote پاورقی فارسی و با دستور

LTRfootnote پاورقی انگلیسی قرار دهید. مثل: در این روش^۵ و یا حسگر^۶. دقت کنید که همه پاورقی‌ها در انتهای متن در بخشی به نام پانویس چاپ خواهند شد و نه در همان صفحه.

در صورتی که می‌خواهید بر روی یک کلمه و یا عبارت در یک متن تأکید کنید، لطفاً از دستور emph استفاده کنید، مثل: کار اصلی ما در این مقاله ارایه یک روش داده کاوی است. تقریباً تمامی بسته‌های مورد نیاز برای نوشتن یک مقاله در استایل L^AT_EX تهیه شده قرار داده شده است. اما در صورت نیاز به یک بسته معین لطفاً استایل قرار داده شده را تغییر دهید و قبل از فراخوانی بسته‌های hyperref و بسته X_qPersian بسته خود را وارد کنید. در هنگام آپلود مقاله نیز کل فایل‌های L^AT_EX باید به صورت zip شده در سایت مورد نظر قرار داده شود. محتویات فایل با پسوند zip باید شامل خروجی pdf، تمامی تصاویر، فایل اصلی tex، فایل با پسوند bib و هر فایل دیگر مورد نیاز باشد.

برای نوشتن فرمول یک خطی در L^AT_EX به سادگی می‌توانید از محیط equation استفاده کنید. مثل:

$$A = B^2 + \frac{\gamma}{4}. \quad (۲)$$

روابط می‌بایست در صورت نیاز حتماً شماره‌گذاری شوند. برای عدم شماره‌گذاری می‌توانید از محیط equation* بهره بگیرید. مثل:

$$A = \frac{\gamma}{\zeta}, \quad \gamma \in \mathbb{R}^{++}.$$

برای روابط چندخطی از محیط align استفاده کنید. مثل:

$$\begin{aligned} A &= B^c + \alpha \\ R &= A + T. \end{aligned} \quad (۳)$$

و یا به عنوان مثال دیگر:

$$\begin{aligned} H(\lambda_g | \lambda_g + \lambda_d) &= \frac{1}{N} \sum_{i=1}^N \log_2(N - i + 1) \\ &+ \frac{1}{N} \sum_{i=1}^N \frac{\sum_{j=i}^N \log_2(\Upsilon_N - \Upsilon_{N-j})}{N - i + 1}, \\ &= \sum_{i=1}^N \frac{\sum_{j=i}^N \log_2(\Upsilon_N - \Upsilon_{N-j})}{N(N - i + 1)}. \end{aligned} \quad (۴)$$

برای ارجاع به روابط و فرمول‌ها نیاز نیست بنویسید مثلاً فرمول شماره ... فقط کافی است که برچسب فرمول را با دستور eqref فراخوانی کنید. در این صورت نیازی به قرار دادن پرانتز نیست و L^AT_EX به صورت خودکار پرانتزها را می‌گذارد. مثلاً با قرار دادن (۲) در (۲) خواهیم داشت ...

سعی کنید مطالب خود را به صورت منظم در قالب تعدادی تعریف، قضیه، لم و گزاره بیاورید. تمامی این محیط‌های در استایل آماده شده تعریف شده است و براحتی قابل استفاده است.

تعریف ۱ (حالت پایدار). حالتی را پایدار گوئیم که در آن تغییر آمارگان پارامترهای صف برابر صفر باشد. اگر ϖ را یک پارامتر صف در نظر بگیریم، خواهیم داشت.

$$\frac{d\varpi(t)}{dt} = 0 \quad (۵)$$

قضیه ۱ (پایستگی جریان). در صورتی که سامانه مورد بررسی ارگودیک و در پایدار باشد، آن گاه نرخ ورود بسته به سامانه همواره برابر با نرخ خروج آن خواهد بود.

قضیه ۲. قرار دادن نام برای قضیه به مانند قضیه قبل که نامش پایستگی جریان بود اختیاری است.

اثبات. سعی کنید اگر قضیه برای مرجع دیگری است اثبات آن را در مقاله بیاورید و فقط به مرجع مذکور ارجاع دهید. اگر اثبات در روند مقاله اهمیت دارد آن را درون متن مقاله بیاورید. اما اگر در روند کلی تأثیری ندارد اثبات را به قسمت پیوست‌ها ببرید. □

نتیجه ۱. در یک صف نرخ ورود با خروج برابر است.

نتیجه ۲. در یک سامانه نرخ ورود برابر است با تعداد بسته‌ها

لم ۱. اگر N_t نشان‌دهنده تعداد بسته رسیده تا زمان t به

$$P(t_i > t) = P(N(t) < i) \implies P(t_i < t) = 1 - \quad (۶)$$

اثبات لم. برای اثبات به مرجع برای اثبات قضایا از محیط proof و برای اثبات لم‌ها از محیط lemmaproof استفاده کنید. ■

اصل ۱ (عدم قطعیت). برطبق اصل عدم قطعیت هر ذره

توصیه می‌شود که نویسندگان برای نمادهای ریاضیاتی سعی کنند از نمادهای ساده و استاندارد استفاده کنند. به عنوان مثال مجموعه اعداد حقیقی را بهتر است به جای R با \mathbb{R} نشان داد. تمامی عملگرهای ریاضیاتی به مانند عملگر امیدریاضی، آنتروپی، احتمال رخداد یک رویداد باید به صورت غیرایتالیک نوشته شود، مثل: $H[\dots]$, $\text{Pr}[\dots]$, $E[\dots]$ و

نویسندگان می‌بایست حتماً و حتماً نمادها و روابط ریاضی موجود در متن را در حالت math mode بنویسند و نه در حالت متنی. به عنوان مثال به جای $A+B$ که به صورت متنی است، بهتر است بنویسید $A + B$.

۷ نتیجه گیری

در این بخش نویسندگان باید به صورت خلاصه کل روندی که در مقاله پیموده شده است را توضیح دهند. در ضمن نویسندگان می‌توانند در این بخش ایده‌های جدید برای توسعه هرچه بیشتر و بهتر مقاله خود را مطرح کنند.

سپاس‌گزاری

بخش سپاس‌گزاری در صورت نیاز بصورت کوتاه و در يك بند آماده شود. بخش سپاس‌گزاری دارای شماره نیست. در این قسمت نویسندگان می‌توانند از افراد و یا نهادهای پشتیبان و یاریگر تشکر و قدردانی کنند. از همین مجال استفاده می‌شود و از همه دوستانی که در روند برگزاری کنفرانس و تهیه این استایل یاریگر ما بودند، تشکر و قدردانی به عمل می‌آید، به خصوص آقایان وفا خیلقی نویسنده بسته ارزشمند X_qPersian، محمد امین طوسی، امیر حسین رضایی تبار، وحید دامن‌افشان، فرشاد ترابی و دیگر دوستان در سایت پارسی‌لاتک.

پیوست‌ها

بخش پیوست‌ها يك بخش اختیاری است و شماره‌گذاری نمی‌شود. موضوع‌های مرتبط با مقاله که در یکی از گروه‌های زیر قرار گیرند، می‌توانند در بخش ضمایم آورده شوند.

- اثبات ریاضی فرمول‌ها یا الگوریتم‌ها.
- داده‌ها و اطلاعات مربوط به مطالعه موردی.
- نتایج کار دیگر محققان و داده‌های مربوط به مقایسه آن‌ها.
- سایر موضوع‌های مرتبط که جزء بخش‌های اصلی مقاله نباشند.

مراجع

- [1] Official *HTML 4.0 Specification*. W3C Recommendation, revised on 24-Apr-1998.
- [2] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. Wiley, 2006.
- [3] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.
- [4] R. T. Durrett. *Essentials of Stochastic Processes*. Springer Texts in Statistics, 2 ed. , 2012.
- [۵] مهاجری، جواد، باباحیدریان، پریسا، و دیانت، ابوالفضل، "طرح جدید مدیریت کلید آستانه ای در شبکه های سیار اقتضایی،" در شانزدهمین کنفرانس ملی سالانه انجمن کامپیوتر ایران، ۱۳۸۹.
- [6] J. Yang and J. Leskovec, "Modeling Information Diffusion in Implicit Networks," *IEEE International Conference on Data Mining*, pp.599–608, Dec. 2010.
- [7] J. Beran, R. Sherman, M. Taqqu, and W. Willinger, "Long-range dependence in variable-bit-rate video traffic," *IEEE Transactions on Communications*, vol.43, no.2, pp.1566–1579, 1995.

پانویس‌ها

^۱read more : [HTML](#)

^۲See also: [Pros of HTML](#)

^۳See also: [Cons of HTML](#)

^۴Anchor Node

^۵در واقع منظور ما ...

^۶Sensor