طراحی نرم افزارهای آموزشی: نگرشی نو به آموزش مبتنی بر رایانه

مسعود کیانی*

چکیده

در زمینه طراحی نرم افزار های آموزشی همواره اختلاف نظر زیادی میان متخصصان آموزش و برنامه نویسان وجود داشته است. هر یک از این دو تلاش نموده اند به گونه ای نظر خود را به دیگری تحمیل کنند. نتیجه چنین اختلاف نظرهایی به وجود آمدن الگوهایی تک بعدی و ناکارآمد در زمینه طراحی و تولید نرم افزارهای آموزشی بوده است. اما باید توجه داشت که طراحی نرم افزار آموزشی فعالیتی تیمی می باشد که همه متخصصان درگیر در آن باید بصورت هماهنگ و به موازات یکدیگر عمل کنند. تنها با در نظر گرفتن اصول فنی و اصول آموزشی در کنار یکدیگر است که می توان نرم افزاری قابل استفاده در کلاس های درس طراحی نمود. درتحقیق حاضر با یک رویکرد توصیفی و تحلیلی به ارائه الگویی جهت طراحی نرم افزار های آموزشی پرداخته شده است. بررسی مفاهیم و اصول نظری طراحی و تولید نرم افزار های آموزشی، ارائه چارچوب ها و الگوهای طراحی و ساخت یک الگوی مناسب طراحی نرم افزار بر مبنای اصول آموزشی از جمله مسائلی هستند که در این مقاله مورد تبیین قرار گرفته اند. بر مبنای الگوی ارائه شده، یک نرم افزار آموزشی کارآمد باید بر مبنای سه محور اصلی تعامل رایانه و دانش آموز، زمینه و سابقه آموزشی و مدیریت اختصاصی مواد یادگیری طراحی شود. در جریان طراحی نرم افزار باید طراح به شناسایی مدل دانش آموزی، نیازسنجی و تدوین هدف های آموزشی، انتخاب و سازماندهی محتوا و طراحی استراتژی های آموزشی بپردازد و ضمن طی فرایند در تمام زمان ها سیستم ایجاد شده را مورد ارزشیابی و در صورت نیاز تجدیدنظر قرار دهد.

واژگان کلیدی

طراحی نرم افزار، نرم افزار آموزشی، تولید نرم افزار، چارچوب های طراحی، متدولوژی های نرم افزار

Keywords: Software Design, Educational Software, Software Development, Design Framework, Software Methodologies

*دانشجوی کارشناسی ارشد دانشگاه خوارزمی (تربیت معلم) تهران و مدرس دانشگاه اصفهان

مقدمه

سیستم های آموزش سنتی سال ها با ارائه اطلاعات در قالب کلاس های درسی و با یک رویکرد منفعلانه به تربیت افراد جامعه پرداخته اند. این سیستم ها به دلیل محدودیت های زیادی که داشته اند، هرگز موفق به برقراری یک نظام هماهنگ و ساختاریافته برای آموزش تمامی افراد جامعه نشده اند. در نظام های آموزشی سنتی بیشتر بر انتقال اطلاعات به درون ذهن دانش آموزان تأکید می شد و کمتر ویژگی های یادگیرندگان و نیازها و خواسته های آن ها مورد توجه قرار می گرفت. اما ظهور رایانه, رشد فناوری اطلاعات و گسترش شبکه های ارتباطی با تغییر در کلیه اجزا نظام جهانی لزوم آموزش پایه برای همه , مهارت های کاری و آموزش مادام العمر را بیش از پیش مطرح کرد و با در اختیار گذاشتن ابزارهای تسهیل انتقال اطلاعات امکان یادگیری مؤثر و مداوم را در بین تمام افراد جامعه فراهم نمود.

یکی از نویسندگان بانفوذی که توان بالقوه‌ی فناوری اطلاعات را به مثابه ابزاری برای ذهنی‌کردن دوباره‌ی فرایند آموزش کشف کرد، سیمور پاپرت[1]بود. پاپرت بر این باور بود که داشتن یک درک عمیق از برنامه‌نویسی منجر به منافع آموزشی چشمگیر و فراوان در بسیاری از قلمروهای بیان، از جمله قلمروهایی که فی‌نفسه ارتباطی با رایانه و فناوری اطلاعات ندارند، می‌گردد. در عین حال وی نسبت به واکنش متعارف جامعه‌ی آموزشی در برابر نیاز به تلفیق بیش از پیش فناوری رایانه در مدرسه از طریق ایجاد آزمایشگاه‌های رایانه و یک زنجیره‌ی یادگیری رسمی شبیه به دیگر دروس انتخابی، ابراز نگرانی می‌کند. پاپرت تأکید می‌کند که هر چند حل‌کردن رایانه در سیستم سنتی، نخستین گام می باشد، اما رایانه در چارچوب قیدوبندهای انعطاف ناپذیر ساختار سنتی مدرسه، ابزار دگرگون کننده‌ای نیست. در عوض، وی براین باور است که دانش‌آموزان می‌توانند قدرت بدیهی رایانه را در دگرگون کردن شیوه‌ی ادراک آموزشی، و نهایتاً پدیدآوردن راه‌های جدید تفکر (فقط از طریق یک محیط اکتشافی آزادتر و انعطاف‌پذیرتر) تجربه کنند. برای اینکه چنین وضعیتی تحقق یابد، باید با بازنگری واصلاح برنامه‌ی درسی؛ رایانه در تمام برنامه‌ی درسی ادغام و حاضر باشد (پاپرت،۱۹۹۹).

باگسترش شبکه های رایانه ای و توسعه و تولید نرم افزار های آموزشی، کشور ها اقدام به توسعه سیستم های باز آموزشی نمودند. گسترش سیستم های باز آموزشی سبب گردید که بسیاری از اقشار جامعه که امکان تحصیل در چارچوب نظام های بسته سنتی را نداشتند اینک بتوانند به تحصیل با استفاده از فناوری های جدید بپردازند. آموزش الکترونیکی در واقع با از بین بردن فواصل مکانی و همچنین وابستگی های زمانی امکان توسعه سیستم های باز را پیش از پیش فراهم ساخت.

آموزش الکترونیکی عبارت از ارائه محتوای آموزشی و تجربیات اساتید مجرب هر رشته از طریق تکنولوژی الکترونیک به دانشجویان علاقه مند در هر نقطه از جهان است. اصطلاح آموزش الکترونیکی، شامل لیست بزرگی از کاربردها و عملکردها از جمله آموزش مبتنی بر رایانه[2] آموزش مبتنی بر وب[3] کلاس های مجازی[4] و همکاری های الکترونیکی[5]می باشد. با آنکه بهره گیری از آموزش

---

[1] Seymour A. Papert

[2] Computer based instruction

[3] Web based training

[4] Virtual classroom

[5] Digital cooperation

مبتنی بر رایانه به دهه ۱۹۷۰ میلادی بر می گردد ولی عملاً تا دهه ۱۹۹۰توجه زیادی به طراحی و توسعه نرم افزار های آموزشی صورت نگرفت. به دنبال بحران دهه ۱۹۷۰ طراحی و تولید نرم افزار های آموزشی به دلایل مختلفی نظیر هزینه بالای طراحی و تولید، عدم توجه به نیازهای مخاطبان، پیشرفت سریع سخت افزار و عدم امکان رقابت نرم افزار با آن، خطاهای فراوان موجود در نرم افزار های تولید شده و نبود امکان نگهداری و پشتیبانی رو به رکود نهاد (حاتمیان, ۱۳۸۲). این رکود تا حدود سال های پایانی دهه ۱۹۸۰ ادامه داشت، ولی با تحولاتی که در این دهه رخ داد لزوم استفاده از فناوری های جدید در جهت بهبود فرایند یاددهی و یادگیری و ایجاد شرایط مناسب یادگیری برای دانش آموزان, استفاده از نرم افزارهای آموزشی را به امری ضروری و اجتناب ناپذیر تبدیل کرد. در این زمان تحقیقات گسترده ای در زمینه طراحی و تولید نرم افزار های آموزشی صورت گرفت. راشل, دیجیانو و پی[۱]در ۱۹۹۸ به اهمیت بکارگیری نرم افزارهای آموزشی در فرایند یاددهی و یادگیری تأکید نمودند. آن ها در تحقیقات خود به این نتیجه رسیدند که نشان گذاری های پویا و پیوندهای چندگانه موجود در نرم افزارهای آموزشی می تواند دانش آموزان عادی را به یادگیری فوق العاده ای مفاهیم ریاضی و علوم قادر سازد.

تقریباً آنچه که در تمامی تحقیقات صورت گرفته در دهه ۱۹۹۰ مشترک بود تأکید آن ها بر طراحی نرم افزارهای آموزشی بصورت تیمی و بر مبنای اصول و ضوابط آموزشی بود. در این دوره محققان بر لزوم همکاری هماهنگ معلمان و برنامه نویسان در قالب تیم های طراحی نرم افزار تأکید نمودند. تا قبل از این طراحی نرم افزار های آموزشی تنها توسط متخصصان برنامه نویسی و تولید نرم افزار صورت می گرفت و معلمان و دانش آموزان نقشی در روند آن نداشتند. این امر باعث شده بود که نرم افزار های طراحی شده عملاً در کلاس های درس به دلیل عدم تطبیق با نیازها و رویکردهای معلمان و دانش آموزان بکار برده نشود.

رعایت اصول طراحی آموزشی در توسعه نرم افزار های آموزشی از اهمیت بسیار زیادی برخوردار است. عدم توجه به تعیین اصول و ضوابط آموزشی و ارائه مدلی در جهت فراهم ساختن شرایط اجرای برنامه درسی در محیط نرم افزار باعث شده که نرم افزارهای آموزشی هرگز نتوانند شرایط آموزش و یادگیری را بهبود بخشند. نرم افزار هایی که با یک الگوی آموزشی معلم مدار و با کمترین حد تعامل با دانش آموز طراحی می شوند, عملکردی مشابه با کتاب های درسی دارند و نمی توانند شرایط را برای بالا بردن سطح یادگیری دانش آموز فراهم سازند. بنابر این لازم است که نرم افزار آموزشی با توجه به اصول نظریه های جدید, نظیر ساخت گرایی, که توجه ویژه ای به عملکرد دانش آموز در محیط نرم افزار دارند, طراحی شود تا ضمن تسهیل یادگیری زمینه را برای توسعه تجارب دانش آموز در یک محیط تعاملی فراهم سازد. در کشور ما متأسفانه به مسأله طراحی نرم افزار های آموزشی تاکنون توجه ویژه ای نشده است و اغلب, طراحی نرم افزار آموزشی توسط مراکز خصوصی و بدون توجه به اصول و چارچوب های تکنولوژی آموزشی صورت گرفته است. همین امر باعث شده که امکان استفاده از این فناوری نوین در مراکز آموزشی ما فراهم نشود و دانش آموزان هنوز در همان سیستم سنتی و با استفاده از کتاب های درسی به تحصیل خود ادامه دهند.

---

[۱]- Roschell, Digiano & Pea

بیان مسأله

آموزش بدون شک یکی از بنیادی ترین مسایل موجود در هر جامعه است, بدین معنی که در توسعه هر جامعه, ارائه یک آموزش بهینه در جهت فرهیخته ساختن افراد آن جامعه امری ضروری می باشد. در سال های اخیر با توجه به اهمیت از بین بردن محدودیت های زمانی و مکانی در آموزش و پرورش و نیاز به توسعه ابزارهایی متحول کننده در سیستم های آموزشی, تأکید بر روی سیستم هایی بوده است که فناوری های هوشمند و تارجهان نما را پشتیبانی کرده اند(رکو، رکو و رکوس[۱]،۲۰۰۴).

مسأله ای که امروز باعث شده است که طراحی نرم افزارهای آموزشی بطور مؤثر صورت نگیرد و فناوری های نوین نتوانند اهداف متعالی آموزشی و پرورشی را پشتیبانی نمایند, این نکته است, که در حقیقت هنوز قوانین و مسیرهای روشنی برای طراحی نرم افزار آموزشی وجود ندارد(هینستروزا و میلار[۲]،۲۰۰۱). دلیل اصلی این موضوع می تواند از یک طرف توجه اندکی باشد که به طراحی نرم افزار های آموزشی شده است و از طرف دیگر آن ممکن است, ناشی از این دیدگاه نادرست باشد که تنها برنامه نویسان با موضوع طراحی نرم افزارهای آموزشی سروکار دارند و آموزشگران نباید در این امر دخالت کنند(میتروپالا و ترایانتی فیلی دیس[۳]،۲۰۰۵).

طراحی نرم افزار های آموزشی توسط برنامه نویسان بدون همکاری و مشارکت معلمان و متخصصان آموزش و پرورش باعث می شود که تنها به اصول فنی طراحی نرم افزار توجه شود و اصول طراحی آموزشی و تئوری های یاددهی و یادگیری مورد غفلت قرار گیرند, بنابر این هنگامی که معلمان می خواهند آموزش خود را با استفاده از نرم افزار آموزشی اجرا کنند با مشکلات زیادی مواجه می شوند(همان، ص۲۸). السون[۴](۲۰۰۰) بر ارتباط بین معلمان و برنامه نویسان در طول طراحی, کاربرد و ارزشیابی نرم افزار آموزشی تأکید می کند. او معتقد است, در مورد هرگونه تغییر و نوآوری, ابتدا معلمان باید تصمیم بگیرند و آن تغییر سپس توسط برنامه نویسان در نرم افزار آموزشی اعمال شود. داشتن چنین دیدگاهی در طراحی و توسعه نرم افزار آموزشی این مسأله را پشتیبانی می کند که برنامه نویسان باید, طراحی نرم افزار را بر مبنای تجارب معلمان صورت دهند.

بنابر این، در طراحی نرم افزارهای آموزشی علاوه بر اصول فنی باید به اصول آموزشی نیز توجه نمود. اصول فنی بیشتر توجه به متدولوژی های طراحی و توسعه نرم افزار بر مبنای مهندسی نرم افزار دارد و اصول آموزشی به فراهم سازی شرایط یادگیری برای گروه های مختلف فراگیران (مخاطب نرم افزار) توجه می کند. براساس این اصول می توان به ساخت یک مدل کارآمد در طراحی نرم افزار آموزشی اقدام نمود. نرم افزار آموزشی باید بتواند نیازهای اساسی موجود در مدارس و یا کلاس های درس را پشتیبانی کند، که این امر مستلزم این است که در جریان طراحی و توسعه نرم افزار معلم و دانش آموز هر دو بصورت فعال مشارکت داشته باشند (میلار و

---

۱. Rokou, Rokou & Rokos

۲. Hinostroza & Mellar

۳. Mitropoulou &Triantafyllidis

۴. Olson

دیگران،۲۰۰۰). در واقع، نرم افزارهای آموزشی باید بتوانند شرایطی را به وجود آورندکه یادگیری دانش آموزان را تضمین نمایند, توجه به اهداف یادگیری, استراتژی های تدریس و سبک های یادگیری می تواند به طور مؤثری از این مسأله پشتیبانی کند (رکو، ۲۰۰۲).

تحقیق حاضر قصد دارد با یک رویکرد توصیفی و تحلیلی به ارائه الگویی جهت طراحی نرم افزار های آموزشی بپردازد. بررسی مفاهیم و اصول نظری طراحی و تولید نرم افزار های آموزشی، ارائه چارچوب ها و مدل های طراحی و ساخت یک الگوی مناسب طراحی نرم افزار بر مبنای اصول آموزشی از جمله اهداف مقاله حاضر است.

پیشینه تحقیق

طراحی نرم افزار آموزشی مسأله ای است که همواره متخصصان تکنولوژی آموزشی به آن توجه داشته اند. فراهم کردن چارچوب های مناسب طراحی, تعیین روش ها و مدل های مطلوب و استفاده بهینه از ابزارهای موجود در جهت توسعه و تولید نرم افزارهای آموزشی از جمله دغدغه هایی هستند که بحث ها و مقالات بسیاری را به خود اختصاص داده اند. میتروپالا و ترایتی فیلی دیس در تحقیقی در سال ۲۰۰۵ به بررسی نگرش معلمان و برنامه نویسان در طراحی و ساخت نرم افزار آموزشی پرداختند، آن ها در تحقیقات خود به این نکته اشاره کردند که وجود اختلاف نظر بین برنامه نویسان و متخصصان آموزشی باعث ایجاد نوعی چالش در زمینه طراحی و تولید نرم افزار های آموزشی شده است, بطوریکه گاه علی رغم طراحی مناسب نرم افزارها از لحاظ اصول برنامه نویسی, آن ها قابل استفاده در عرصه های آموزشی توسط معلمان نیستند. کاشی هیرا[۱](۲۰۰۳) در کتاب خود با عنوان طراحی مؤثر محیط های آموزشی مبتنی بر رایانه, به نقش رایانه به عنوان یک رسانه آموزشی اشاره می کند و به بحث در رابطه با یادگیری مبتنی بر رایانه در تئوری و عمل می پردازد. او می گوید در طراحی نرم افزار های آموزشی باید به روابط بین انسان و رایانه و همچنین تئوری های یادگیری توجه نمود. در نظر گرفتن عوامل انسانی, قابلیت کاربرد, طراحی رابط کاربر,انتخاب زبان فرمان, بهره گیری از فرامتن ها, نگاره ها و انیمیشن ها و سازماندهی و تحلیل وظیفه از دیگر عواملی هستند که کاشی هیرا در طراحی نرم افزارهای آموزشی مورد بحث و بررسی قرار می دهد. میهالکا[۲](۲۰۰۵) به طراحی نرم افزار آموزشی برای ریاضیات دوره ابتدایی توجه کرده است. او ضمن توجه به مفاهیم شیء گرایی و ساخت گرایی در آموزش, به بحث در رابطه با طراحی نرم افزار آموزشی پرداخته, او همچنین چارچوب هایی برای ارزیابی نرم افزار آموزشی ارائه کرده است.

بلتون و بیس[۳]به بررسی نرم افزار مهندسی سیستم های شناختی پرداخته اند. آن ها معتقدند که یکی از کمبودهای اساسی موجود در تکنولوژی آموزشی توجه اندک آن به عناوین مهندسی سیستم های شناختی از جمله مدل هایی از عملکرد انسان در محیط های پویا است. توجه به مدل سازی، ارزشیابی و طراحی ابزارهای برنامه درسی بصورت یکپارچه بر مبنای آن ها, قضاوت و تصمیم گیری انسانی

---

۱ Kashihara

۲ Mihalca

۳ Bolton & Bass

را در محیط های نرم افزاری تسهیل می کند. نرم افزار مهندسی سیستم های شناختی اسناد ومعلمان هوشمند را شامل می شود و همچنین طراحی انعطاف پذیر در جهت افزودن محیط های تصمیم گیری و قضاوت جدید به سیستم را تسهیل می کند. لیج, زبنکو و کتادی[۱] (۲۰۰۱) یک چرخه را برای طراحی نرم افزار آموزشی پیشنهاد کرده اند, این چرخه شامل امکان سنجی, تعریف نیازهای سیستم, تصریح نیازهای نمونه اولیه, طراحی نمونه اولیه, طراحی جزئیات نمونه اولیه, توسعه نمونه اولیه, اجرا نمونه اولیه و آزمون, طراحی سیستم نهایی, اجرا سیستم نهایی, بهره برداری و نگهداری و اصلاح و تجدیدنظر می باشد. اسکورز[۲](۱۹۹۹) در طراحی نرم افزار آموزشی توجه ویژه ای به ساخت گرایی دارد و به پیشنهاد راهبردها و ملاک هایی برای تدوین نرم افزار, توسعه محیط های آموزشی تعاملی و کاربرد ساخت گرایی در سیستم های فناوری اطلاعات و ارتباطات مدرن از جمله تار جهان نما و شناسایی فنون آموزشی نوین می پردازد. دوبل[۳](۲۰۰۲) معتقد است در زمینه نرم افزار آموزشی باید بتوان به چند سئوال اساسی پاسخ داد: آیا نرم افزار نمونه های روشنی از مهارت هایی را که برای توسعه آن ها طراحی شده فراهم می کند؟, آیا نرم افزار اجازه آموزش اختصاصی را می دهد؟, آیا نرم افزار مسیرهایی را برای بهبود پیشنهاد می کند و توانایی تنظیم خودکار برای انطباق با نیازهای دانش آموزان را دارد؟, آیا نرم افزار می تواند مانند یک معلم واقعی به دانش آموزان بازخورد دهد یا فقط می تواند نشان دهد که پاسخ دانش آموزان درست یا غلط است؟و... . راشل, دیجیانو و پی (۱۹۹۸) درتحقیقات خود به این نتیجه رسیدند که تکنیک های گذشته در طراحی نرم افزار های آموزشی محصولات گران, ناسازگار و انعطاف ناپذیر نتیجه داده است. اکثر پروژه های نرم افزاری که در بخش دولتی تولید شده اند اگر چه توانسته اند نتایج امیدوار کننده ای در آزمون های کوچک از خود نشان دهند ولی هرگز نتوانسته اند بازار گسترده ای را برای خود فراهم سازند. این محققان معتقدند که در طراحی نرم افزار آموزشی باید روشی را اتخاذ کرد, که بتوان برمبنای آن محصولی با کیفیت بالا در آموزش علوم و ریاضیات تولید نمود. بیبر[۴](۲۰۰۵) معتقد است که توسعه دهندگان محتوای آموزشی به ابزار های ساخت یافته , متدولوژی ها و راهبرد های ویژه در تدوین نرم افزار آموزشی مؤثر نیاز دارند. بیبر چارچوبی را برای توسعه ابزارها, متدولوژی ها و راهبرد های مؤثردر طراحی نرم افزار های آموزشی پیشنهاد می دهد.

در سال های اخیر در زمینه طراحی نرم افزار های آموزشی تحقیقات گسترده ای صورت گرفته است, که از جمله این تحقیقات می توان به تحقیقات کیپر, هوگ وجانگ[۵](۲۰۰۱), بیل و شارپ لس[۶](۲۰۰۲), محمد احمد[۷](۲۰۰۳), ناتس[۸](۲۰۰۴), ریگ بی[۹](۲۰۰۵)

---

۱ Lage, Zubenko & Cataldi

۲ Squires

۳ Deubel

۴ Bieber

۵ Kuyper , Hoog & Jong

۶ Beale &Sharples

۷ Muhammad Ahmed

۸ Notess

۹ Rigby

اشاره کرد. در ایران متأسفانه تاکنون تحقیقات گسترده ای در زمینه طراحی نرم افزار های آموزشی صورت نگرفته است و تحقیقات موجود نیز بسیار ناکامل می باشد.

محورهای طراحی نرم افزار

ارائه یک الگوی واحد که بتواند در طراحی و تولید تمامی نرم افزار های آموزشی کاربرد داشته باشد کاری بسیار دشوار است. طراحی هر نرم افزار با توجه به درون دادها، فرایند و برون دادهای مورد نظر مستلزم طرح ریزی چارچوبی ویژه می باشد که باید توسط تکنولوژیست آموزشی تدوین شود. نرم افزار های آموزشی را می توان با توجه به نوع کاربردشان به دودسته کلی تقسیم نمود. یک دسته نرم افزارهایی که جهت کمک به فرایند یاددهی و یادگیری و در واقع تسهیل آن، به همراه معلم در کلاس درس مورد استفاده قرار می گیرند. این دسته از نرم افزارها به عنوان یک ماده کمک آموزشی وظیفه تکمیل و پشتیبانی از مسئولیت های آموزشی معلم در کلاس درس را برعهده دارند. دسته دوم نرم افزار هایی هستند که به عنوان یک رسانه آموزشی واحد جهت توسعه یادگیری مستقل مورد استفاده قرار می گیرند. این دسته از نرم افزار ها به گونه ای طراحی می شوند که نقش معلم را برای دانش آموز بازی می کنند. پاسخ دادن به سئوالات دانش آموز، همراهی او در طی مرور درس، ارائه بازخورد سریع، فراهم سازی شرایط آسایش فکری و ارتباط دو طرفه از جمله ویژگی های مطلوب این دسته از نرم افزارها محسوب می شود. با بهره گیری از نرم افزارهای آموزشی خودآموز، دانش آموزان از قید مکان و زمان رها می شوند و می توانند با توجه به علایق خود یک یادگیری انفرادی و مستقل را تجربه کنند.

برای طراحی و تولید نرم افزار های آموزشی باید سه اصل مهم و محوری را مورد توجه قرار داد. تعامل دانش آموز و رایانه، زمینه و سابقه آموزشی دانش آموز و مدیریت انفرادی مواد یادگیری این سه اصل را تشکیل می دهند. در زمینه **تعامل دانش آموز و رایانه**، باید نرم افزار بتواند شرایطی را فراهم کند که دانش آموز به راحتی با برنامه ارتباط برقرار کرده و یک یادگیری فعال را تجربه کند. برای اینکه تعامل دانش آموز و رایانه در یک سطح بالایی حفظ شود، باید طراحی به صورتی باشد که توجه دانش آموز در روند یادگیری به شکل محیط و فضای ایجاد شده جلب نشود، بلکه بر روی مواد یادگیری متمرکز گردد. توصیه می شود که شکل محیط به گونه ای باشد که دانش آموز در آن احساس راحتی کرده و آنچه را که روی صفحه نمایش می بینند به آسانی قابل خواندن و ادارک باشد (وین[1] ۱۹۹۳).

هنگامی که از نرم افزار در کلاس درس استفاده می شود، معلمان گرایش دارند شرایطی به وجود آورند تا دانش آموزان با نرم افزار برای دستیابی به یادگیری ارتباط برقرار کرده و یک یادگیری فعال را تجربه کنند (کاتسوریس و گیالیس[2] ۱۹۹۴). برای همین منظور، متخصصان باید در جریان طراحی نرم افزار علاوه بر سطح تعامل دانش آموزان و فعالیت هایشان در ارتباط با محتوا، استراتژی ای که

---

[1] Winn

[2] Katsoris & Gialis

معلم به منظور ارائه محتوا بکار خواهد گرفت را نیز طراحی کنند. عدم توجه به روش ها و فنون تدریس باعث می شود که امکان ایجاد تعامل بین دانش آموز و رایانه فراهم نشود. در واقع معلمان در صورتی که وظیفه خود را در هنگام استفاده از نرم افزار ندانند ترجیح می دهند به همان روش سنتی به تدریس خود ادامه دهند. این در حالی است که بهره گیری از نرم افزار در کلاس مستلزم رعایت اصول و ضوابط خاصی می باشد که باید در طراحی آن لحاظ شود. تعامل بین دانش آموز و رایانه، علاقه و توجه او را در طول مرور محتوا در سطح بالایی حفظ می کند. فراهم کردن انتخاب های مداوم, پاسخ دادن به سئوالات یا وظایف قرار داده شده و دریافت بازخورد فوری در مورد درست و یا غلط بودن عملکرد از جمله مواردی هستند که علاقه دانش آموزان را در زمان استفاده از نرم افزار آموزشی در سطح بالا نگه می دارند. (هینستروزا و میلار،۲۰۰۱).

توجه به **زمینه و سابقه آموزشی دانش آموزان** جهت طراحی یک نرم افزار مؤثر ضروری است. در واقع، طراح برای اینکه دانش آموزان را در روند استفاده از برنامه به هدف های آموزشی مورد نظر برساند، باید دقیقاً بداند که نرم افزار برای چه کسانی طراحی می شود. الگوها و فرایند های ذهنی انسان, راه های ادراک و دریافت از استراتژی های آموزشی و نیز تئوری های یاددهی و یادگیری از جمله مواردی هستند که با استناد به سابقه و زمینه های قبلی دانش آموزان بدست می آیند (رکو، ۲۰۰۲).

برای طراحی یک نرم افزار آموزشی بر مبنای زمینه و سابقه آموزشی (ویژگی های یادگیرنده, سبک های یادگیری و روشهای یاددهی) دانش آموزان، باید مدلی مناسب را برگزید. یکی از شناخته شده ترین مدل ها در حوزه علوم شناختی مدل بلوم[۱] است. در این مدل سطح شناختی فراگیر به شش سطح دانش, درک و فهم، کاربرد، تجزیه و تحلیل، ترکیب و قضاوت و ارزشیابی تقسیم شده و ویژگی های مربوطه و تناظرات آموزشی هر سطح تبیین می شود. در فرایند مدل سازی سیستم, مدل باور, خواسته و نیت (BDI[۲]) همراه با سبک های یادگیری فراگیر استخراج می شود که در این بین مدل شناختی بلوم می تواند بعنوان مرجعی در بخش مدل سازی برای ایجاد یادگیری در سطوح مختلف مورد استفاده قرار گیرد . مدل BDI یکی از مدل های حوزه شناختی است که تعابیر آن در حوزه فراگیری عبارتند از :۱) **نیت؛** معرف انگیزه و تمایل فراگیر در فرایند یادگیری است که سطح شناختی مطلوب در برخورد با مسائل از جمله مصادیق این تمایلات می باشد.۲) **خواسته؛** مبین درخواست فراگیر از سیستم یا فرایند فراگیری, و ۳) **باور،** دانسته ها و توانایی های کاربر است که سنجیده یا ادعا می شود. علاوه بر توجه به این ویژگی ها باید به شیوه های یادگیری نیز در ساختار آموزش مبتنی بر نرم افزار توجه شود، مشخصه های شناختی, انگیزشی و فیزیولوژیکی شخص فراگیر که بعنوان معیاری قابل سنجش از نحوه ادراک و تعامل او با محیط می باشند باید به طور دقیق در مراحل گوناگون ساختاردهی به آموزش مدنظر قرار گیرند(عبدحق، ۱۳۸۲).

در مدل سازی شیوه های یادگیری بر اساس سوابق و زمینه های دانش آموزان می توان از چند تئوری بهره برد. تئوری های دان[۳],کلب[۴], میرز و بریگز[۵]از جمله این تئوری ها می باشند. در تئوری دان تمرکز اصلی شیوه های گوناگون تحریکی, در فرایند مدل سازی و

---

۱ Bloom

۲ Belief, Desire and Intention

۳ Dunn & Dunn

فراگیری است (هین و بودنی[3] ۲۰۰۱). در تئوری کلب که یکی از معتبرترین تئوری های حوزه شناختی است با بررسی زنجیره پردازش فراگیر(در یک بازده از کاملاً فعال تا کاملاً منفعل) و همچنین شیوه ادراک او (در یک بازده کاملاً انتزاعی تا کاملاً پیوسته) فراگیران را به گروه های مختلف طبقه بندی می کند. در مدل میرز و بریگز کاربران بر اساس ویژگی های برون گرایی, احساس, بینش, تفکر و قضاوت طبقه بندی می شوند(تندی و جیسر[4] ۱۹۹۸).

در زمینه **مدیریت انفرادی مواد یادگیری،** ما باید مجموعه ای از نقش ها و مکانیسم ها را در انتخاب مواد یادگیری مبتنی بر ویژگی های دانش آموزان, اهداف آموزشی, روش های تدریس و رسانه های در دسترس قصد کنیم که شرایط را برای یک یادگیری انفرادی فراهم سازند. بالا بردن سطح تعامل سیستم با دانش آموزان و همچنین توجه به زمینه و سوابق دانش آموزان می تواند در فراهم سازی شرایط لازم برای مدیریت انفرادی مواد یادگیری راه گشا باشد.

ساخت الگویی برای طراحی نرم افزار

الگوی آموزشی تئوری ها, اصول و فرایندهایی که هدفشان استاندارد کردن فرایند آموزش و دسته بندی مجدد متدولوژی هاست را شرح می دهد. درست همانطور که یک مهندس ساختمان پیش از ساختن یک بنا باید به تهیه نقشه آن بپردازد، طراح نرم افزار نیز باید یک نقشه آموزشی طراحی کند. این نقشه طی فرایندی تعاملی تجزیه و تحلیل و طراحی می شودکه در آن محتوا، راهبردهای آموزشی و رسانه های مناسب انتخاب، مرتب و استفاده می گردند. فرایند تعریف یک مدل آموزشی در شکل زیر نشان داده شده است.

شکل۱. فرایند تعریف یک مدل آموزشی

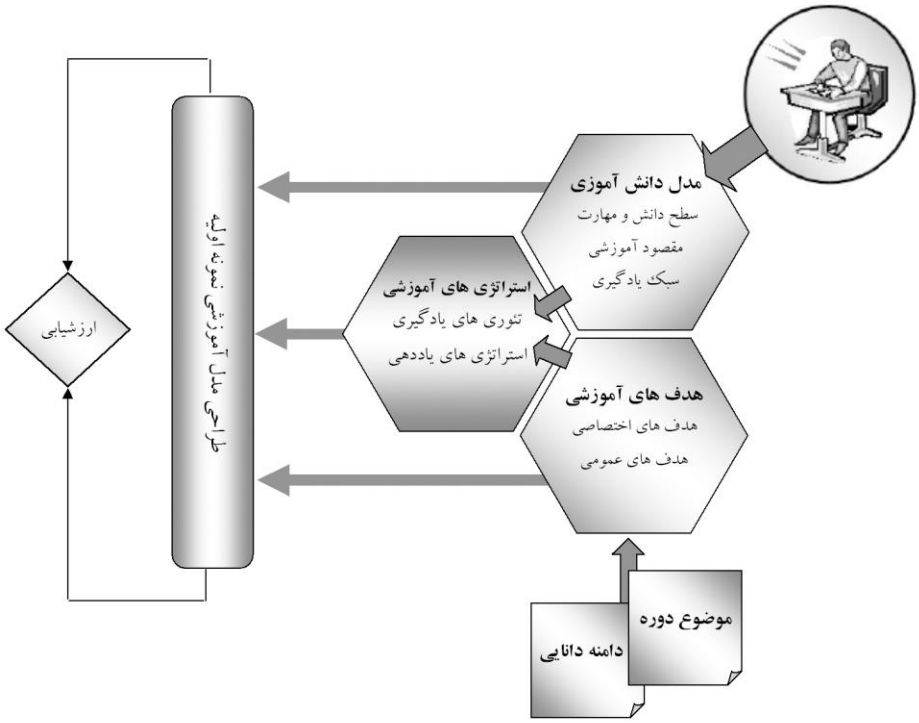
---

[1] Kolb

[2] Myers & Briggs

[3] Hein & Budny

[4] Tendy & Geiser



منبع: (رکو، رکو و رکوس،۲۰۰۴)

چنانکه در شکل می بینید فرایند تعریف یک مدل آموزشی از سه مرحله اصلی تعریف ویژگی های دانش آموزان، تعریف هدف های آموزشی و تعریف استراتژی های آموزشی تشکیل شده است.

تعریف ویژگی های دانش آموزان

انتخاب یک مدل مناسب برای طرح ریزی نرم افزار آموزشی بیش از هر چیز مستلزم شناسایی دانش آموزان مخاطب آن می باشد. همانطور که در بحث «محورها» ذکر شد، یکی از محورهای اصلی در زمینه طراحی نرم افزار های آموزشی توجه به زمینه و سابقه دانش آموزان می باشد. برای اینکه شما بتوانید یک مدل کامل را تنظیم کنید باید بطور دقیق به جمع آوری داده هایی در زمینه سبک های یادگیری، مهارت و دانش دانش آموزان بپردازید. همچنین توجه به ملاحظاتی نظیر سن، مرحله و سطح کلاسی دانش آموزان نیز در طی فرایند طراحی باید در نظر گرفته شود. براساس نظر آموزشگران درجه دشواری هر درس باید مطابق با سن، سطح رشد شناختی و توانایی دانش آموزان کلاس باشد، بر این مبنا نرم افزار باید مطابق با سطح معینی از دشواری طراحی شودتا دانش آموزان بتوانند با یک روند از ساده به دشوار مراحل آموزشی را یکی پس از دیگری طی نمایند(رکو، ۲۰۰۲).

انتخاب مدل مناسب با توجه به ویژگی های دانش آموزان کار دشواری نیست. اگر شما به فعالیت های معلمان در سر کلاس های درسی سنتی توجه کنید، ملاحظه می نمایید که آن ها از طریق گرفتن آزمون، مطالعه پرونده دانش آموزان و پاسخ های آن ها در سر کلاس اقدام به ساخت فرضیاتی در مورد یک یک دانش آموزان می کنند و سپس آموزش خود را براساس همین فرضیات پیاده سازی می نمایند. طراح نرم افزار آموزشی نیز با هوشمندسازی برنامه و ایجاد امکان مدیریت آن، شرایط را برای شناسایی ویژگی های

دانش آموزان فراهم می آورد. ایجاد آزمون های ورودی و مرحله ای و همچنین فراهم سازی فهرستی از مشخصات و ویژگی های خاص هر دانش آموز می تواند طراح را در امر تولید نرم افزار یاری کند.

برای تهیه یک «مدل دانش آموزی» کامل لازم است طراح یک سری از مشخصات و ویژگی ها را فهرست کند. رابرت میگر[۱](۱۹۹۷) در مورد تحلیل یادگیرنده رویه های زیر را توصیه می نماید:

۱. با درک اینکه تحلیل یادگیرنده یک سند کاری است، که بوسیله هیچ کس دیگری جز خودتان و شاید دیگر اعضای تیم طراحی ایجاد نمی شود آغاز کنید.
۲. هر چیزی را که فکر می کنید به شما در شناخت مخاطبان مورد نظر کمک می کند یادداشت کنید.
۳. دامنه ویژگی های یادگیرندگان را تا جایی که امکان دارد توصیف نمایید.

در مجموع، برطبق نظریه میگر، وقتی شما تحلیل یادگیرنده را اجرا می کنید، باید اطمینان داشته باشید که موارد زیر را انجام می دهید:

- مراقب باشید یادگیرندگان را چنان توصیف کنید که آن ها هستند، نه آنچنان که شما می خواهید باشند.
- افراد را توصیف کنید، نه سازمان ها یا خط مشی ها را.
- تفاوت های یادگیرندگان و نیز شباهت های ایشان را در نظر بگیرید.
- به خاطر داشته باشید که شما یک سند کاری ایجاد می کنید، که می تواند در طول فرایند تولید اصلاح یا ترمیم شود.

دیک، کری و کری[۲](۲۰۰۱) معتقدند در جریان تحلیل یادگیرنده طراح باید اطلاعاتی در زمینه رفتار ورودی، دانش قبلی در حیطه موضوع، نگرش نسبت به محتوا و پتانسیل سیستم انتقال، انگیزه تحصیلی، سطح توانایی و آموزش، رجحان های یادگیری متداول، نگرش نسبت به سازمان ارائه دهنده آموزش و ویژگی های گروهی دانش آموزان جمع آوری کند.

هاینیک، مولندا، راسل و اسمالدینو[۳](۲۰۰۲) دیدگاه متفاوتی را ارائه کرده اند. آن ها پیشنهاد کردند که تحلیل یادگیرنده باید بر روی سه جنبه مخاطب مورد نظر متمرکز باشد: ویژگی های عمومی، صلاحیت های ورودی ویژه و سبک های یادگیری.

- **ویژگی های عمومی:** ویژگی های عمومی، اطلاعات جمعیت شناختی، از قبیل سن یادگیرنده، توانایی های فیزیکی، و موقعیت اجتماعی او؛ تعیین هویت فرهنگی یا ارزش های مشترک یادگیرنده، و مقداری از تجارب قبلی را که بوسیله محتوا پوشش داده شده است، شامل می شود. این اطلاعات می تواند به طراح درباره این که چگونه آموزش را در قالب نرم افزار تدارک ببیند یا ارائه دهد کمک کند.
- **صلاحیت های ورودی ویژه:** دانش قبلی فراگیران، که در موفقیت آن ها ضروری است، در کامیابی پروژه طراحی نرم افزارآموزشی بسیار حیاتی می باشد. نداشتن آگاهی از دانش قبلی دانش آموزان باعث می شود که برنامه طراحی شده نتواند

---

۱ Robert F. Mager

۲ Dick, Carey & Carey

۳ Heinich, Molenda, Russell & Smaldino

نیازهای آن ها را برآورده سازد. به عنوان مثال اگر دانش آموزی از مفاهیم پایه ای مانند جمع و تفریق آگاهی نداشته باشد نمی توان در قالب نرم افزار بدون پرداختن به این دو عمل اصلی از ضرب و تقسیم سخن به میان آورد.

- **سبک های یادگیری:** رجحان های روانشناختی اشخاص معلول این است که آن ها آموزش را جذاب و اثربخش بیابند. جنبه های هوش گاردنر[1](۱۹۹۳) (کلامی- زبانشناختی، منطقی- ریاضی، بصری- فضایی، موسیقایی- مسجع، بدنی- جنبشی، میان فردی، طبیعت گرا و وجود گرا) و "سبک های ذهن" گریگور[2](محسوس- ترتیبی، محسوس- تصادفی، انتزاعی-ترتیبی، و انتزاعی- تصادفی) مثال هایی هستند برای تبیین اینکه مردم چگونه نظریه ای را ترجیح می دهند و اقدام به حل مشکلات می نمایند (بوتلر[3] ۱۹۸۶). اطلاع از این رجحان ها ممکن است به طراح کمک کند، آموزشی ایجاد نماید که مخاطب مورد نظر بتواند به آسانی درک کند و بپذیرد.

در زمینه یادگیری، یکی از مفروضات بنیادی این است که همه اشخاص می توانند چالش های یادگیری را به روش های مختلفی حل و فصل نمایند. لیکن این راه حل ها لزوماً فردی نیستند؛ آن ها می توانند در سبک های مشترک طبقه بندی شوند. آرتور الیس[4](۲۰۰۱) این مسأله را اینگونه توضیح می دهد:

> اکثریت سبک ها حامی این دیدگاه هستند که هر یک از ما اطلاعات را بطور متفاوتی دریافت و پردازش می کنیم و بخاطر این مسأله، معلمان باید کوشش زیادی بکنند تا دریابند که دانش آموزان چگونه به بهترین وجه یاد می گیرند. منطق این تفکر به ما دیکته می کند که اگر چه سبک های یادگیری متفاوتند، آن ها برابر هستند. این استدلال می گوید که هوش و توانایی هم عرض هستند لیکن در میان انسان ها بطور متفاوتی توزیع شده اند. تکالیف مدرسه نوعاً به ایجاد تبعیض، با طرفداری یا ضدیت با دانش آموزان معین، گرایش دارند. مثلاً سخنرانی ممکن است واقعاً مناسب سبک یادگیری یک شخص (برای مثال، سبک شنیداری) نسبت به دیگران باشد. این مسئله از جهت دیگری اشاره دارد بر اینکه توانایی افراد در بسیاری موارد نادیده گرفته می شود، نه به این دلیل که آن ها ناتوان از یادگیری اندیشه ها هستند بلکه به این خاطر که راه حل یادگیری نامناسب است (ص ۱۵۱).

آگاهی از اینکه یادگیرندگان اطلاعات را به روش های مختلفی ادراک می کنند، باید به یادآورد که مسئول طراحی نرم افزار آموزشی باید تعدادی از فعالیت ها را فراهم نماید که بتواند تفکر یادگیرندگان را به روش های مختلف شبیه سازی نماید.

تعریف هدف های آموزشی

یکی از مقدماتی ترین فعالیت ها در اغلب الگوهای طراحی نرم افزار، سنجش نیازهاست. هر آموزشی باید پاسخ گوی یک سلسله نیازها باشد که در صورت اجرای صحیح آموزش، آن نیازها رفع شوند. در واقع نیاز به موقعیتی دلالت دارد که در آن وضعیت موجود یا حاضر با وضعیت مطلوب فاصله دارد (فتحی واجارگاه، ۱۳۸۱). وضعیت مطلوب یا آنچه که ایده آل و ترجیحات را شکل می دهد،

---

۱ Gardner

۲ Gregorc

۳ Butler

۴ Arthur Ellis

همان هدف ها هستند، زیرا بنا به تعریف، هدف ها؛ غایات و نتایج نهایی حاصل از آموزشند که نرم افزار آموزشی به منظور دستیابی به آن ها طراحی و تولید می شود. نتایج نهایی، همان توانایی های پایانی فراگیران اعم از دانسته ها، مهارت های عملی و گرایش های مختلف ناشی از آموزش هستند که ارائه آموزش در قالب نرم افزار را توجیه و تفسیر می کند. نداشتن هدف و شروع کار طراحی نرم افزار به معنی شروع سفری بدون مقصد، بدون راه و بدون نقشه و قطب نماست، یا به عبارتی شروع سرگردانی است.

طراحان نرم افزارهای آموزشی پیش از هر چیز باید بدانند که مخاطبان آن ها قبل از اینکه گروهی از یادگیرندگان باشند، انسان هایی هستند که خواست ها و نیازهای خاص خود را دارند. اگر چه در نظر گرفتن همه نیازهای انسانی وظیفه طراح نرم افزار نیست، ولی آگاهی از میزان آمادگی دانش آموزان جهت بهره گیری از نرم افزار امری ضروری می باشد. بسیاری از محققان معتقدند که نرم افزار آموزشی باید بتواند بر سطوح مختلف نیازها تأکید کند و زمینه رفع آن ها را با ارائه مواد آموزشی مناسب و فراهم سازی امکان مشارکت، خودباوری و تصمیم گیری در فرایند فراهم سازد.

مزلو[1](۱۹۶۸) نیازهای انسانی را در چهار طبقه نیاز به سلامت فیزیکی، عاطفی، ذهنی و معنوی طبقه بندی کرده و معتقد است رفتارهای اشخاص به وسیله نیازهای آن ها تعیین می شود. سلسله مراتب نیازهای مزلو بصورت هرمی رسم شده که در قاعده آن آسایش های فیزیکی از قبیل رفع گرسنگی و تشنگی، در مراتب بالاتر نیازهای انتزاعی تر مانند دانش، زیبایی شناسی و خود شکوفایی و در رأس آن توانایی کمک به دیگران جهت دستیابی به خودشکوفایی و تعالی قرار گرفته است (ارلیچ و دیگران[2] ۲۰۰۴).

شکل ۲. سلسه مراتب نیازهای مزلو

---

[1] Maslow

[2] Orlich & et al

منبع: (برون و گرین[1] ۲۰۰۵)

در طراحی نرم افزارهای آموزشی باید به نیازهای مختلف انسانی دانش آموزان توجه نمود و زمینه را برای دستیابی آن ها به سطوح بالاتر فراهم ساخت. میلر, پرستون, هین استروزا و ربین[2](۲۰۰۰) معتقدند که در طراحی نرم افزارهای آموزشی، نیازهای اساسی موجود در مدارس و یا کلاس ها مورد ملاحظه قرار نمی گیرد و عوض آن، طراحی مبتنی بر ایده های شخصیِ طراح نرم افزار است. آن ها از مشارکت معلمان و دانش آموزان در جریان طراحی نرم افزار آموزشی حمایت کرده و بر طراحی نرم افزار بر مبنای نیازهای موجود در مدارس تأکید می کنند.

پس از تعریف نیازها نوبت به تدوین هدف های آموزشی می رسد. هدف ها بطور عمده بر نوع و موضوع مواد آموزشی و بر مهارت ها و توانایی هایی که ما می خواهیم دانش آموزان بدست آورند مبتنی هستند (پانتنو رکو[3] ۲۰۰۲). هدف های آموزشی را بطور کلی می توان به دو دسته زیر تقسیم نمود:

- **هدف های اختصاصی.** این هدف ها مربوط به موضوع درس و یا موضوعی واحد و ویژه از مواد آموزشی می شوند. برای مثال اگر شما بخواهید نرم افزاری در زمینه آموزش ریاضیات دوره ابتدایی طراحی کنید، لازم است که ابتدا براساس

---

[1] Brown & Green

[2] Mellar, Preston, Hinstroza & Rehbein

[3] Pantano Rokou

سرفصل ها و محتوای کتاب های درسی و مواد آموزشی ریاضیات دوره ابتدایی، هدف ها را برگزینید. این هدف ها اختصاصی و مربوط به همان دوره می باشند.

- **هدف های عمومی.** این هدف ها مبتنی بر مهارت ها و توانایی هایی هستند که شما انتظار دارید دانش آموزان در فرایند تحصیل به آن ها دسترسی پیدا کنند. مهارت های حل مسأله، تفکر انتقادی، پژوهش، سنجش و ارزشیابی از جمله این مهارت ها می باشند. نرم افزارهای آموزشی باید به گونه ای طراحی شوند که به رشد و توسعه این مهارت ها و توانایی ها در دانش آموزان کمک کنند.

طراحان نرم افزار در نوشتن هدف های آموزشی باید دقیقاً توجه داشته باشند که توانایی های نهایی بصورت دقیق مشخص و بیان شوند. تمیز هدف های اختصاصی و عمومی نیز از امور قابل ملاحظه است که در تهیه نرم افزار آموزشی باید مدنظر قرار گیرند.

تعریف استراتژی های آموزشی

برای تعریف یک مدل کارآمد جهت طراحی نرم افزارهای آموزشی، علاوه بر شناسایی مدل دانش آموزی و تدوین اهداف آموزشی، باید به تعریف استراتژی های آموزشی مناسب اقدام نمود. شناسایی استراتژی های آموزشی بر مبنای ویژگی های دانش آموزان و محتوای دوره صورت می گیرد. در نظر گرفتن استراتژی های آموزشی در امر طراحی و تولید نرم افزار، بار دیگر بر این مسأله تأکید می کند که طراحی نرم افزار کاری تیمی است که با همکاری معلمان صورت می گیرد. طراحان نرم افزار باید تلاش کنند که نیازهای معلمان و کلیه عواملی را که مرتبط با استراتژی های آموزشی آنان می باشد بهتر درک کنند و ضمن تأکید بر دانش و تجارب معلمان، طراحی خود را بر آنچه که آن ها می توانند با نرم افزار انجام دهند متمرکز نمایند (السون، ۱۹۸۸).

یکی از نکاتی که در شناسایی استراتژی آموزشی مناسب می تواند به طراحان نرم افزارهای آموزشی کمک کند، آشنایی با تئوری های یادگیری است. این تئوری ها مبنایی را برای تعریف فرایند یادگیری فراهم می کنند و می توانند طراح را در انتخاب استراتژی مناسب یاددهی براساس مدل دانش آموزی یاری نماید. امروزه محققان در طراحی نرم افزارهای آموزشی بر تئوری ساخت گرایی تأکید می کنند. تئوری ساخت گرایی به عنوان یک رویکرد جدید در فلسفه تربیتی در اواخر قرن نوزدهم و اوایل قرن بیستم بر مبنای نظریات روانشناسی پیاژه[۱]و برونر[۲]و دیدگاه فلسفی دیویی[۳] کانت[۴]و کوهن[۵]شکل گرفت. در واقع ساخت گرایی یک روش فکری درباره چگونه دانستن است و یک مبناء برای ساخت استراتژی های آموزشی و برنامه درسی محسوب می شود (توبین وتیپینز[۶] ۱۹۹۳).

---

۱ Piaget

۲ Brunner

۳ Dewey

۴ Kant

۵ Kuhnt

۶ Tobin & Tippins

تئوری ساخت گرایی به توصیف ابعاد مختلف آموزش می پردازد و با ایجاد یک دیدگاه تربیتی جامع، طراح را در تعریف مدلی پویا و کارآمد یاری می نماید. در جدول زیر به مقایسه دو رویکرد سنتی (عینیت گرایی) و رویکرد جدید (ساخت گرایی) پرداخته شده است.

جدول ۱. ابعاد آموزش از رویکرد سنتی به رویکرد جدید

طبقات	دوکرانه پیوستار(از رویکرد سنتی به رویکرد جدید)	
معرفت شناسی	عینیت گرایی	ساخت گرایی
فلسفه آموزشی	معلم گرا	ساخت گرا
اصول روانشناسی	رفتارگرایی و شناخت گرایی	ساخت گرایی
توالی آموزش	کاهش گری	ساخت گرایی
جهت گیری هدف ها	متمرکز	غیر متمرکز
نقش معلم و آموزشگر	قدرت طلب و یاددهنده	تساوی طلب و تسهیل کننده
ارزش تجربه کردن	بی مسمی	محسوس و آشکار
انعطاف پذیری برنامه	مقاوم در برابر معلم	به آسانی تغییرپذیر
ارزش خطاها	یادگیری بدون خطا	یادگیری از تجارب
انگیزش	دارای منبع خارجی	دارای منبع درونی
ساخت یافتگی	بالا	پایین
کنترل یادگیرنده	وجود ندارد	نامحدود
فعالیت کاربر	منفعل	فعال
انطباق با تفاوت های فردی	وجود ندارد	ازهمه جهت
یادگیری مشارکتی	عدم پشتیبانی	جدایی ناپذیر
حساسیت فرهنگی	وجود ندارد	جدایی ناپذیر

اقتباس از:(ریوز[۱] ۱۹۹۴و ریوز و هارمن[۲] ۱۹۹۴)

در آموزش بر مبنای تئوری ساخت گرایی بر یادگیری موقعیتی، فعال بودن شاگردان در فرایند آموزش، شاگرد محوری، یادگیری اکتشافی, تکالیف واقعی و اصیل مرتبط با موقعیت های واقعی زندگی، یادگیری مشارکتی، استفاده از سطوح بالای فعالیت های شناختی نظیر حل مسأله، استدلال و تفکر انتقادی، انگیزش با منبع درونی و ایجاد ارتباط بین دانش قبلی و جدید یادگیرندگان تأکید می شود (کیانی، ۱۳۸۵). بهره گیری از تئوری یادگیری ساخت گرا در جریان طراحی نرم افزارهای آموزشی مستلزم توجه به ویژگی های فوق است.

اگرچه در رویکرد عینیت گرا سازمان و توالی محتوا بدون توجه به نیازها و علایق دانش آموزان از پیش تعیین شده است، ولی در نرم افزارهای آموزشی مبتنی بر رویکرد ساخت گرا از این توالی خبری نیست. دانش آموزان در این نرم افزارها اجازه پیدا می کنند هر

---

۱ Reeves

۲ Harmon

بخش از محتوا را که می خواهند با توجه به نیاز و علاقه اشان مطالعه کنند. همچنین هوشمند سازی نرم افزارهای آموزشی و فراهم سازی امکان مدیریت محتوای آن ها، این امکان را به معلمان داده است که محتوای نرم افزار را باتوجه به ویژگی های دانش آموزان و روش تدریس شان اصلاح و یا ترمیم نمایند. این بدین معنی است که معلمان قادر خواهند بود مواد آموزشی اولیه را که توسط طراح نرم افزار در آن قرار گرفته تغییر دهند(جیرینا[1] ۱۹۹۲). مرور آزاد محتوا و همچنین تطبیق آن با نیازها و شرایط دانش آموزان باعث می شود علاقه آن ها به محتوای نرم افزار حفظ شود. محققان نشان داده اند که حفظ کردن علاقه دانش آموزان یکی از عواملی است که باید بطور جدی در جریان طراحی نرم افزار آموزشی مورد ملاحظه قرار گیرد(شلبرن، گشن و هاکسلی[2]، ۲۰۰۱).

فعال بودن دانش آموزان در جریان آموزش از اصول بنیادی تئوری ساخت گرایی است. یک نرم افزار آموزشی باید به گونه ای طراحی شود که این اطمینان را بدهد که دانش آموزان در روند آموزش تنها تماشاگرانی منفعل نیستند. مقصود از فعال بودن فراگیر تنها اقدام به فشار یک دکمه به منظور پیشروی از یک صفحه به صفحه دیگر نیست؛ بلکه هدف این است که دانش آموزان در یک جریان تعاملی در آموزش درگیر شوند و در کل فرایند با اقدام به مشکل گشایی، حل مسأله و ارزشیابی فعالیت نمایند. بهره گیری از چندرسانه ای ها، همراه بودن صدا با متن، استفاده از فرامتن ها و فرا پیوندها، پرهیز از تکرار مکرر اطلاعات از جمله مواردی هستند که می توانند درگیر بودن دانش آموزان در روند آموزش را تضمین کنند.

بهره گیری از چند رسانه ای ها در آموزش می تواند بر جذابیت آن بیافزاید و دانش آموزان را در موقعیتی عینی تر قرار دهد. دانش آموزان معمولاً علاقه مند هستند که مواد آموزشی در قالب فیلم های ویدیویی، انیمیشن ها و تصاویر متحرک به آن ها ارائه شود. بهره گیری از این نوع چندرسانه ای ها علاوه بر درگیر کردن فراگیران در فرایند آموزش، مشکل خواندن متن از روی صفحه نمایش را نیز برطرف می کند. محققان به پژوهش هایی در زمینه بررسی و مقایسه میزان کارآمدی خواندن متن از روی صفحه نمایش و از روی رسانه های چاپی (نظیر کتاب) اقدام نموده اند. نتایج این تحقیقات نشان داد که در دوساعت مطالعه مداوم، خواندن متن از روی صفحه نمایش ۲۸٪ آهسته تر در مقایسه با خواندن متن چاپی از کتاب است. این محققان توصیه نمودند که به منظور بهبود کارآمدی مطالعه از صفحه نمایش، متن در شکل های مختلف و یا همراه با سایر رسانه ها به دانش آموزان ارائه شود(هانگ و لین[3]، ۲۰۰۱). **همراهی صدا با متن** نیز می تواند در بهبود کارآمدی خواندن از صفحه نمایش و همچنین درگیر نمودن دانش آموزان با محتوا نقشی محوری داشته باشد. در هنگام استفاده از گوینده، باید مراقب بود که آنچه توسط او بیان می شود با متن انطباق کامل داشته باشد. در غیر این صورت فراگیر دچار سردرگمی شده و نمی تواند اطلاعات را دریافت کند (مک کرلی و پریس[4] ۱۹۹۳).

بهره گیری از فرامتن ها و فراپیوندها می تواند طراح را در ایجاد آموزشی مبتنی بر رویکرد ساخت گرایی یاری کند. فرامتن ها می توانند با فراهم سازی امکان دسترسی فوری دانش آموزان به فهرست لغات فنی، اطلاعات تکمیلی و تعاریف، آن ها را از سردرگمی در هنگام مطالعه نجات دهند. از این نظر خواندن کتاب های چاپی می تواند خسته کننده باشد. خواننده در حین مطالعه کتاب برای دسترسی به اطلاعات تکمیلی باید به فهرست راهنما و یا ضمیمه ها رجوع کند که این امر باعث پاره شدن رشته تفکر و تمرکز او می شود (کاتسوریس و گیالیس، ۱۹۹۴). فراپیوندها یک محیط یادگیری غنی را برای فراگیران فراهم می کنند. تجربه نشان

---

[1] Giardina

[2] Shelbourn, Ghassan & Hoxley

[3] Huang & Lin

[4] McKerlie & Preece

داده است که بهره گیری از فراپیوندها در قالب نرم افزارهای آموزشی، جهانی واقعی تر از آنچه در محیط یاددهی و یادگیری سنتی فراهم شده ایجاد می کند (دارمپا، کردروی و هنگر[1] ۲۰۰۰).

برخی از معلمان تصور می کنند که تکرار مکرر اطلاعات در قالب نرم افزار آموزشی می تواند به یادگیری دانش آموزان کمک کند. همانطور که می دانید برمبنای تئوری ساخت گرایی، یادگیری فرایند فعال ساخت دانش است. تکرار اطلاعات باعث می شود دانش آموزان میل به حفظ کردن طوطی وار اطلاعات پیدا کنند. در واقع تکرار مکرر اطلاعات نه تنها نتیجه مطلوب در بر ندارد بلکه به تدریج باعث کاهش علاقه و انگیزه دانش آموزان به یادگیری می شود (هانگ و لین، ۲۰۰۱). دانش آموزان معمولاً از توجه به اطلاعاتی که برای بار دوم و سوم بصورت خودکار ارائه می شود خودداری می کنند. تکرار تنها وقتی می تواند مفید باشد که یادگیرندگان با توجه به نیازشان و بصورت هوشیارانه آن را درخواست کنند. محققان معتقدند که طراحان نرم افزارهای آموزشی بهتر است به جای تأکید بر تکرار مکرر اطلاعات، شرایط را برای ارائه آن ها در یک زمان طولانی تر و در طی گام هایی کوتاه تر فراهم نمایند.

بحث و نتیجه گیری

طراحی یکی از مهمترین مراحل در چرخه تولید نرم افزارهای آموزشی می باشد. متخصصان تکنولوژی آموزشی، طراحان و برنامه نویسان تاکنون چارچوب های بسیاری را برای تعریف الگویی کارآمد جهت طرح ریزی نرم افزارهای آموزشی ارائه کرده اند، اما هیچ یک کامل و تأمین کننده نیازهای معلمان در مدارس نبوده اند.

برخی از متخصصان فرایند طراحی نرم افزارهای آموزشی را به دو مرحله طراحی محتوا و طراحی و برنامه نویسی نرم افزار تقسیم می کنند. آن ها معتقدند که ابتدا باید محتوا براساس نیازسنجی صورت گرفته و اهداف تعیین شده طراحی شود و سپس در قالب نرم افزار قرار گیرد. داشتن چنین دیدگاهی سبب می شود که ما نتوانیم در فرایند طراحی از حداکثر تسهیلات و امکاناتی که یک نرم افزار در اختیار می گذارد بهره مند شویم. طراحی نرم افزار یک کار تیمی است که باید بصورت یکپارچه و هماهنگ صورت گیرد. در واقع باید دو مرحله طراحی محتوا و طراحی نرم افزار را درهم ترکیب کرد تا بتوان نرم افزاری کارآمد طراحی و تولید نمود.

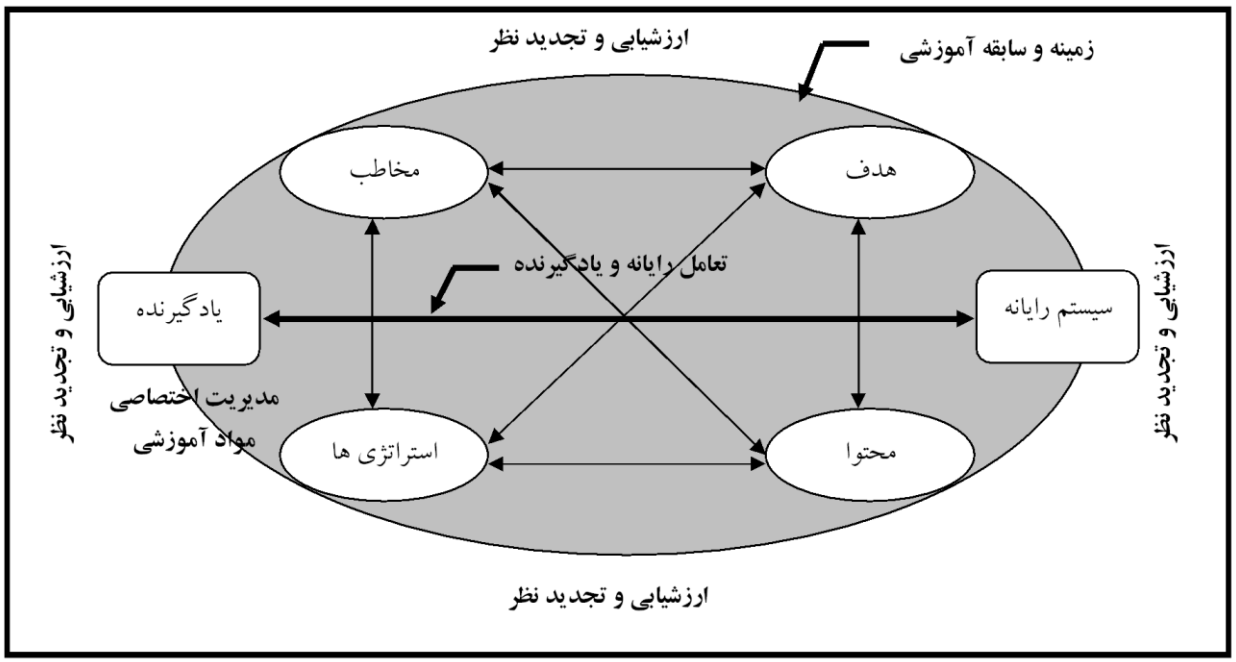
برای تعریف یک الگوی مناسب جهت طراحی نرم افزار های آموزشی باید یک سری از مؤلفه ها را در نظر گرفت. همانطور که ذکر شد، یک نرم افزار آموزشی باید با توجه به سه محور اصلی **تعامل رایانه و دانش آموز، زمینه و سابقه آموزشی و مدیریت اختصاصی مواد یادگیری** طراحی و تولید شود. این سه محور تضمین کننده کارایی و اثربخشی نرم افزار می باشند. در نظر نگرفتن هر یک از این محورها باعث می شود که دانش آموزان و معلمان در جریان استفاده از نرم افزار دچار اشکال شوند. تعامل رایانه و دانش آموز تضمین کننده برقراری یک ارتباط کامل بین یادگیرنده و مواد آموزشی می باشد. دانش آموزان برای دستیابی به یادگیری باید بتوانند به تعامل با مواد آموزشی پرداخته و ضمن انجام فعالیت هایی نظیر حل مسائل درسی، پاسخ به سئوالات موردی، رفع اشکالات و انجام تکالیف عملیاتی، یاد بگیرند. دستیابی به یادگیری تنها زمانی ممکن است که دانش آموزان در جریان تعامل با رایانه با مواد آموزشی مواجه

[1] Dharmappa, Corderoy & Hangare

شوند که پیش زمینه لازم برای بهره گیری از آن ها را دارند. عدم توجه به سابقه و زمینه آموزشی دانش آموزان سبب می شود که آن ها با محتوایی ناآشنا و نامفهوم روبرو شده و نتوانند از مواد آموزشی بهره مند شوند.

علاوه بر تعامل رایانه و دانش آموز و زمینه و سابقه آموزشی، برای فراهم سازی شرایط یادگیری مستقل و انفرادی باید اصل مدیریت اختصاصی مواد آموزشی را لحاظ نمود. مدیریت اختصاصی مواد آموزشی به این معنا نیست، که نرم افزار به گونه ای طراحی شود که نتوان آن را در کلاس درس همراه با معلم و یا سایر رسانه های آموزشی بکار برد. بلکه این محور بر این نکته تأکید می کند که دانش آموزان باید بتوانند نرم افزار را بدون کمک دیگران و در هر زمانی که نیاز داشتند برای توسعه دانش و مهارت های خود مورد استفاده قرار دهند. در واقع مدیریت اختصاصی مواد آموزشی تضمین کننده اصل عدم وابستگی یادگیرندگان به زمان و مکان می باشد.

شکل ۳. الگوی طراحی نرم افزار های آموزشی



پس از اینکه محورهای اصلی طراحی نرم افزار شناسایی شدند، لازم است نرم افزار در چهار مرحله اصل **شناسایی مدل دانش آموزی (مخاطب)، نیازسنجی و تعریف هدف های آموزشی، انتخاب و سازماندهی محتوا و طراحی استراتژی های آموزشی** مدل سازی شود. اضافه نمودن مرحله انتخاب و سازماندهی محتوا به سه مرحله قبلی از این رو صورت گرفت که علی رغم اینکه با مشخص شدن محورها و اهداف آموزشی انتخاب و سازماندهی محتوا تا حد زیادی تسهیل می شود، ولی فرایند انتخاب و سازماندهی محتوا همواره از حساسیت و اهمیت ویژه ای برخوردار بوده است. این حساسیت و اهمیت نیز بیشتر به علت ماهیت تصمیماتی است که درباره محتوا اتخاذ می شود.

از آنجا که طراحی و تولید یک نرم افزار امری زمان بر و مستلزم هزینه مالی فراوان می باشد باید بتوان همواره آن را مورد تجدید نظر و اصلاح قرار داد. هزینه تولید یک نرم افزار مناسب با گرافیک بالا و تعامل برخط می تواند از چند صد دلار تا میلیون ها دلار می باشد. اگرچه طراحان در مراحل اولیه تمام محورها و مؤلفه ها را جهت طراحی نرم افزار در نظر می گیرند، ولی باید همواره ارزشیابی و تطبیق نرم افزار با جدیدترین نیازهای معلمان و دانش آموزان مدنظر قرار گیرد. نرم افزار آموزشی خوب نرم افزاری است که مطابق با نیازهای معلمان و دانش آموزان، تسهیل کننده فرایند آموزش، قابل کاربرد در زمینه ها و برنامه های متعدد، قابل اصلاح و تجدیدنظر و قابل مدیریت باشد. اگرچه ساخت چنین نرم افزاری به نظر دشوار می رسد ولی غیرممکن نیست.

منابع و مآخذ

حاتمیان، ر. (۱۳۸۲). *مهندسی نرم افزار*. تهران: مؤسسه فرهنگی هنری دیباگران تهران.

عبدحق، ب و خراط، م. (۱۳۸۳). *رویکردهای گوناگون در حوزه فراگیری، ویژگی های محیط های آموزشی، تقسیم بندی موضوعات آموزشی و بررسی سبک های فراگیری*. گزارش داخلی گروه کاربرد فناوری اطلاعات، تهران: پژوهشکده فناوری اطلاعات و مرکز تحقیقات مخابرات ایران.

فتحی واجارگاه، ک. . (۱۳۸۱). *اصول برنامه ریزی درسی*. تهران: ایران زمین.

کیانی، م. (۱۳۸۵). آموزش ساختارگرایانه: رویکردی نوین در آموزش و پرورش. *مجموعه مقالات نهمین هفته پژوهش دانشگاه اصفهان*. اصفهان:انتشارات دانشگاه اصفهان.

Ahmed, M. I. (2003). *A Practical Process for Reviewing and Selecting Educational Software.* Indiana: PLATO Learning, Inc.

Beale, R., & Sharples, M. (2002). *Design Guide for Developers of Educational Software.* BECTA.

Bieber, M. (2005). *Infrastructures for Future Educational Software.* New Jersey: New Jersey Institute of Technology.

Bolton, M. L., & Bass, E. J. (2005). Cognitive Systems Engineering Educational Software (CSEES): Educational Software Addressing Quantitative Models of Performance. *IEEE International Conference on Systems* (pp. 10-12). Hawaii : Man and Cybernetics.

Brown, A. H., & Green, T. D. (2005). *The essential of instructional design: connecting fundamental principles with process and practice.* New Jersey: Prentice Hall.

Butler, K. A. (1986). *Learning and teaching style: in theory and practice* (2nd ed.). Columbia: CT: Learne'r dimension.

Deubel, P. (2002). Selecting curriculum based software. *Learning & Leading with Technology , 29* (5), 25-30.

Dharmappa, H., Corderoy, R., & Hangare, P. (2000). Developing an interactive multimedia software package to enhance understanding of and learning outcomes in water treatment processes. *Journal of Cleaner Productions* (8), 408.

Dick, W., Carey, L., & Carey, J. O. (2001). *The systematic design of instruction* (5th ed.). New York: Addison-Wesley/ Longman.

Ellis, A. K. (2001). *Research on educational innovations* (3rd ed.). New York: Eye on education.

Giardina, M. (1992). Interactivity and intelligent advisory strategies in a multimedia learning environment: Human factors, design issues and technical considerations. *NATO ASI Series F : Computer and Systems Sciences , 93*.

Hein, T. L., & Budny, D. D. (2001). Research on learning style: Application in the physics and Engineering Classrooms. *IEEE Trans. On Education , 44* (3), 33-36.

Heinich, R., Molenda, M., Russell, J., & Smaldino, S. (2002). *Instructional media and technologies for learning* (7th ed.). New Jersey: Merrill/Prentice Hall.

Hinostroza, J. E., & Mellar, H. (2001). Pedagogy embedded in educational software design: report of a case study. *Computers and Education , 37*, 27.

Huang, S. T., & Lin, M. H. (2001). Designing efficient text presentation of multimedia CAI – the evaluation of dynamic text patterns and the negative repetition effect on memory. *Computers & Education , 37*, 128.

Kashihara, A. (2003). Book Review: The Knowledge Medium: Designing Effective Computer Based Learning Environments (Author: G. A. Berg). *Educational Technology & Society , 6(3)*, 84-85.

Katsoris, P., & Gialis, K. (1994). Multimedia Educational Association in Cell Biology. *8th EUCEN Conference: Using Multimedia Methods in University Continuing Education* (p. 75). Greece: the International Olympic Academy.

Kuyper, M., Hoog, R. d., & Jong, T. d. (2001). *Modeling and supporting the authoring process of multimedia simulation based educational software: a knowledge engineering approach.* Amsterdam: the EC in its Telematics programme as project ET1020.

Lage, F. J., Zubenko, Y., & Cataldi, Z. (2001). an extended methodology for educational software design: some critical points. *31th ASEE/IEEE Frontiers in Education Conference.* Reno.

Mager, R. F. (1997). *Making instruction work, or skillbloomers* (2nd ed.). Atlanta: CA: Center for effective performance.

McKerlie, D., & Preece, J. (1993). The hype and the media: Issues concerned with designing hypermedia. *Journal of Microcomputer A__lications , 16*, 39.

Mellar, H., Preston, C., Hinstroza, J. E., & Rehbein, L. (2000). Developing educational software: A professional tool perspective, education & information technologies. *Kluwer Academic Publishers , 5* (2), 103-117.

Mihalca, L. (2005). Designing Educational Software for Learning Mathematics in Primary Education. *Recent Research Developments in Learning Technology* , 20-22.

Mitropoulou, V., & Triantafyllidis, G. (2005). Design of Educational Software: Programmer and Teacher Approach. *3rd International Symposium of Interactive Media Design*, (pp. 1-9). Istanbul.

Notess, M. (2004). Applying Contextual Design to Educational Software Development. *a view from the teach* , Idea Group Publishers.

Olson, A. (1988). *Schoolworlds/Microworlds: Computers and the Culture of the Classroom.* Oxford: Pergamon Press.

Orlich, D. C., Harder, R. J., Callahan, R. C., Trevisan, M. S., & Brown, A. H. (2004). *Teaching strstegies: A guide to effective instruction* (7th ed.). Boston: Hounghton mifflin.

Pantano Rokou, F. (2002). A pedagogical model for distance education with the use of information and communication technologies. *Asian journal of information technology , 2* (1), 8-12.

Papert, S. A. (1999). *Mindstorms: Children, Computers, and Powerful Ideas* (Second ed.). New York: Basic Books.

Reeves, T. C. (1994). *Evaluating what really matters in Computer-Based Education.* Retrieved 2007, from Australia's leading Information and Communications Technology agency : http://www.educationau.edu.au/archives/cp/reeves.htm.

Reeves, T. C., & Harmon, W. (1994). Systematic evaluation procedures for interactive multimedia for education and raining. In S. Reisman, *Multimedia computing: Preparing for the 21st century.* Harrisburg: PA: Ida Group.

Rigby, C. P. (2005). The challenges of creating open source educational software: the Gild experience. *Proceedings of the First International Conference on Open Source Systems*, (pp. 334-338). Geneva.

Rokou, F. (2002). *Distance Learning with Hypermedia Technologies: Design pedagogical Model and communication process.* Athens: Kritiki (in Modern Greek).

Rokou, F. P., Rokou, E., & Rokos, Y. (2004). Modeling Web-based Educational Systems: Process Design Teaching Model. *Educational Technology & Society , 7 (1)*, 42-50.

Roschelle, J., Digiano, C., Pea, R., & Kaput, J. (1998). Educational Software Components of Tomorrow (ESCOT). *SRI International* , 1-13.

Shelbourn, M., Ghassan, A., & Hoxley, M. (2001). Multimedia in construction education: new dimensions. *Automation in Construction , 10*, 265-274.

Squires, D. (1999). Educational Software and Learning: Subversive Use and Volatile Design. *Proceedings of the 32nd Hawaii International Conference on System Sciences.* Hawaii .

Tendy, S. M., & Geiser, W. F. (1999). The Search for Style: IT all Depends on Where you Look. *Net Forum Teacher Educ , 9* (1), 25-27.

Tobin, k., & Tippins, D. (1993). Constructivism as a referent for teaching and learning. In K. Tobin, *The Practice of Constructivism in Science Education* (pp. 3-21). New Jersey: Lawrence- Erlbaum, Hillsdale.

Winn, W. A. (1993). A Constructivist Critique of the Assumptions of Instructional Design. *Proceedings of the NATO Advanced Research Workshop in Belgium 1991* (p. 206). Berlin: Springer-Verlag.