Designing Simulations for use in Higher Education

William Schoenberg and Michael Bean

Forio Online Simulations
1159 Howard Street
San Francisco, CA 94103
(415) 440 7500
bschoenberg@forio.com, mbean@forio.com

Introduction:
There is a popular notion that the best designs for products, buildings, and software are produced by a powerful and inspired visionary while poor designs emerge from democratized groups (Grossman, 2002). While the notion of a dictatorial Steve Jobs driving great design is appealing, it is incorrect.

"Design by dictator" works well for projects where the requirements are straight-forward, the penalties for mistakes are minimal, and stakeholder buy-in is inconsequential. "Design by committee" works well when projects are quality driven, requirements are complex, consequences of error are serious, or stakeholder buy-in is essential (Lindwell, 2003).

Design is improved when developed by a team with diverse members and member input and contributions are efficiently collected and shared and a simple governance model is adopted to facilitate decision making and ensure that the design cannot be delayed or deadlocked (Laughlin, 2003). The same principles that hold for designing products, buildings and software also hold true for simulations. In this paper, we explore how designs of simulation by a diverse team of stakeholders can be efficiently implemented.

Understanding the simulation development process requires an understanding of the definition of simulation. For the purposes of this paper a simulation is defined as the combination of a mathematical model and a user interface (UI). The mathematical model is developed based on a variety of modeling techniques based on the subject matter being taught. The user interface contains a set of screens for both students and faculty. The development of a model that yields key insights relative to the learning objectives of the exercise is only part of designing a successful simulation product. It is important to design an engaging student user interface experience and a simple and easy to use set of administration and result gathering screens for faculty.

To ensure a consistent product quality we have developed a process which can be applied to simulation development whether web-based, desktop-based, or mobile-based. The simulation development process starts as soon as a client approaches us about developing a simulation. The process can be broken down into two general phases: design and development. Each phase is based on an agile/iterative development process with iteration meetings scheduled at regular intervals (Larman, 2004). The iterative approach was chosen as the basis for the development process because of its proven benefits in the software development field (Chow and Cao, 2008, Larman, 2004). This is especially true for projects where the final products are unknown at the time of project scheduling, as well as when there is an expectation for scope to be clarified as the project progresses (Chow and Cao, 2008, Larman, 2004).

The process makes a distinction between design and development as a way to manage the risk inherent in all product development; especially in simulation based product development. Each project phase is budgeted for separately; using a different iteration schedule in order to optimize the productivity of the simulation designers and developers based on their expected tasks. Project roles and responsibilities shift as the project moves from the design phase to the development phase.

Roles in the Simulation Development Process
At our organization, a typical simulation project will have five different roles filled by at least two, and up to five people. The five roles are: Project Manager, Modeler, User Interface Designer, User Interface Developer, and Subject Matter Expert. Each role plays a specific part in the simulation design process and participates at different stages of design or development.

Project Manager
The project manager is responsible for the day to day operation and coordination of work on the project. They work closely with all other project members to ensure that iteration goals are being met, and no team member is unnecessarily waiting for another member to complete a task. The project manager is responsible for documenting the project development and for communicating project status with the client, and key stakeholders not taking part in the actual development of the project. The project manager is heavily involved in the project through both the design and development phases.

Modeler
Modelers are responsible for developing the mathematical model at the heart of the simulation. They work most closely with subject matter experts to understand the concepts being taught and the pedagogical objectives of the simulation exercise, and translate those concepts and goals into a simulation. Modelers typically are trained in system dynamics, discrete event simulation, or agent based simulation. Modelers specialize along specific topics such as finance, operations research, or pricing etc. in order to minimize the amount of time it takes for the modeler to gain fluency in a particular topic. Modelers are heavily involved in the project during the design phase and early in development. Typically their role in the project diminishes after the initial iterations of the development phase.

User Interface Designer
User interface designers are responsible for coming up with the visual experience for the simulation by providing wireframes, mockups, and functional specifications of the user interface. User interface designers use their knowledge of web-design, graphic design, typography etc. to design an intuitive and visually pleasing user interface for both students and faculty based on the input of the subject matter expert, project manager and modeler. User interface designers are most active in the project during the design phase, and as the project transitions into the development phase they take a smaller role.

User Interface Developer
User interface developers are responsible for translating the simulation design into a working product. The user interface developer role is filled by a programmer with knowledge of web interface technologies such as HTML, CSS, and JavaScript with an ability to develop websites that work across a variety of popular web browsers. User interface developers initially work in close proximity to user interface designers and modelers to make sure that the code that they develop matches the design and reveals the results and key insights of the model. The user interface developer is only involved in the project's development phase.

Subject Matter Expert
Subject matter experts for simulations developed for use in higher education are typically university faculty. They are responsible for guiding the design of the simulation. Subject matter experts:

1. Make sure that the model reveals key insights around the specific teaching objectives.
2. Give advice around the design of the student portion of the user interface in order to maximize usability and engagement.
3. Inform the design of the facilitator user interface based on their experience as a faculty. Using this experience they are able to guide the user interface designer to make sure only the relevant and necessary information is available to faculty.
4. Produce a reference guide for the simulation which describes how the simulation should be taught and introduced so that other faculty who wish to use the simulation as a part of their curriculum are able to have a starting point for integrating the simulation into their class.

The subject matter expert is most involved in the design portion of the project, and near the end of development in order to verify that the final product meets their needs, and the needs of the end-users.

The Design Phase
The design phase is the first of two phases in the development process. During the design phase no user interface code is written, only wireframes, mockups, functional

specifications and work on the mathematical model is completed. The reason for this is to allow the simulation team to explore and map out the full scope of the simulation project. During this phase of simulation development, the iterations are kept as short as possible to maintain momentum and to keep a coherent vision of the simulation in the forefront of the simulation development team's minds.

The first meeting in the design phase is called the project kickoff before anything is known about the content of the simulation besides a topic. In order to accommodate this lack of knowledge around the requirements of the project by the client we engineer the kickoff meeting to generate a high-level scope-map of the project.

The goal of the kickoff meeting is to not only map out the potential scope for the project, but to develop an initial working causal loop diagram of the underlying simulation model. The brainstorming process requires a facilitator and that all key client stakeholders and any subject matter experts be present. Typically the group includes around ten people, and the process can be run successfully with between five and fifty people. The process starts with the facilitator listing an open-ended question. This question serves as a starting point for the subject matter experts to begin discussing the simulation. Each statement relevant to the simulation mentioned by the participants is recorded and placed on a wall (Figure 1). This portion of the process is the most divergent, allowing the participants to explore the full potential scope of the simulation. Participants will typically discuss topics ranging from very low level specific details about the process being modeled, to general ideas about user interface design, to key learning objectives and goals for the simulation.

Figure 1: This is a sample brainstorming session with ideas formed into clusters identifying the major areas to be covered by the simulation

After the initial brainstorming the facilitator groups related ideas together. The facilitator then asks the participants for their suggestions around the grouping of remaining ideas.

The next step in the process is for the facilitator to prompt the participants to name each cluster of ideas. This process forces the participants to think about their categorizations and to further refine and organize their ideas. Generally as a part of this process the participants will move some ideas from one cluster to another as different participants have different ideas about what the contents of each cluster were specifically.

The final part of the process is to take the variables generated from the groups and to organize them into causal links. Gradually the facilitator guides the participants into building a full scale causal loop diagram by pointing out specific areas where there may be feedback or delays at play. After the process is complete the scope of the simulation, and the basis for the model have been established enough to begin the rest of the design phase.

Shortly after the kickoff meeting the project manager, modeler and subject matter expert work closely together to generate a key list of learning objectives. These learning objectives will form the base on which the entire simulation will be developed.

The purpose of generating the learning objectives is to focus the efforts of the simulation design phase. These learning objectives will form the basis for the simulation, and will be the metrics by which all changes to the simulation are based. Each change to the simulation will be scored according to how it either positively or negatively affects the learning objectives.

Two weeks after the kickoff meeting the first of the design phase iterations is held (iteration #1). At this meeting the project manager, subject matter expert and user interface designer are present. The purpose of this meeting is to discuss wireframes (Figure 2) which have been assembled by the user interface designer based on the results of the kickoff meeting. Wireframes are black and white, low fidelity images which are used to figure out how the simulation content is organized across various pages, and then for each page how in the most general sense the content is to be organized and presented to users. A good analogy for this process is sketching. At this first iteration meeting the simulation development team presents to the client the initial wireframes with the goal of garnering feedback from the various participants on how to revise and change the layout and organization of the simulation.

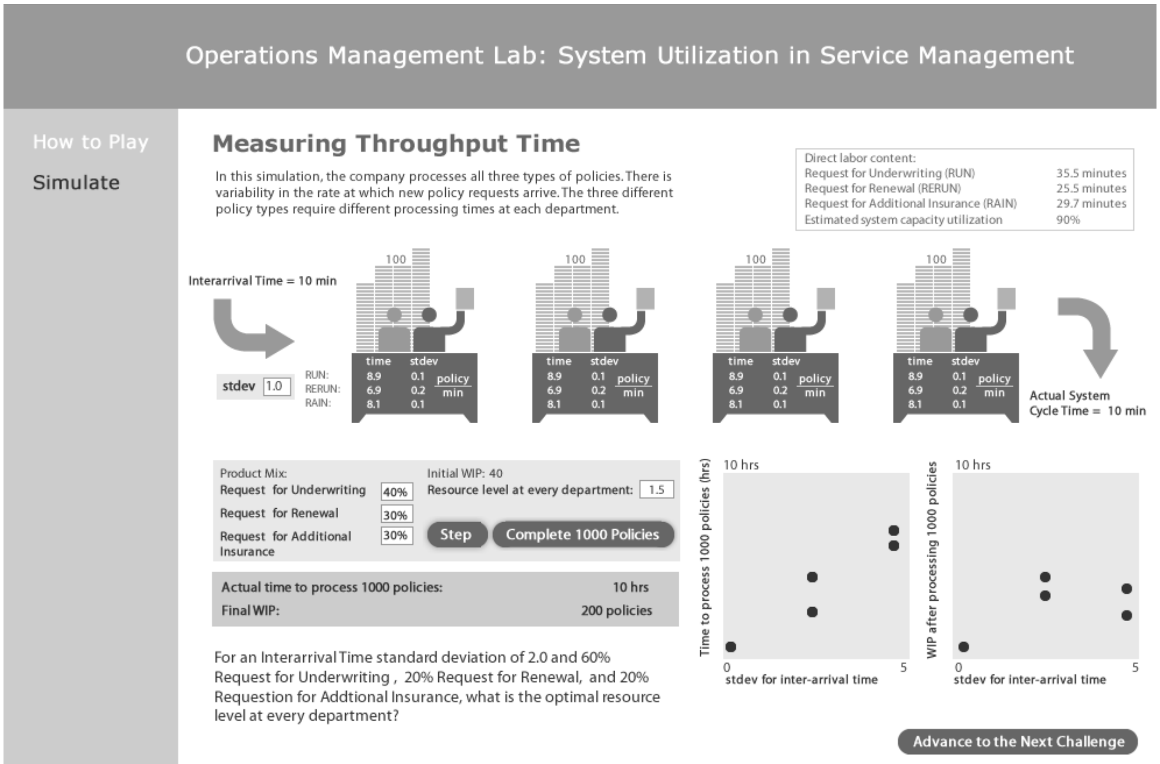

Figure 2: This is a sample wireframe from a simulation project. In it you can see that the much attention is paid to page layout, but little attention is paid to details surrounding the user interface elements, and there is no color applied.

A week after the first wireframes are presented, the simulation design team meets again for iteration #2. They review any updates made to the wireframes, based on feedback from the previous meeting, as well as discuss a model prototype developed by the modeler. Model work is included as a part of the simulation design phase, as opposed to the simulation development phase because changes to the model have a strong impact on user interface design and development. Therefore if model development is pushed off to the development phase of the project, there is a much higher likelihood of rework and wasted effort. The modeler shares the results of their model using causal loop diagrams, stock and flow diagrams, as well as spreadsheets of preliminary model results. It is important to have a running model at this point in time so that the subject matter expert can begin to understand the practical implications of their theories when represented with the precision that mathematical simulation requires.

Two weeks after the revised wireframes are presented the team meets again to discuss the initial mockups (Figure 3) for the simulation. Mockups are high fidelity, full color, representations of the screens that will be developed during the simulation development phase. These screens need to be reviewed carefully for content, design (adherence to visual design guidelines), ease of use, and consistency across product lines. Once these mockups have been agreed upon by the simulation development team they will be the specification used by the user interface developer to construct the simulation code.

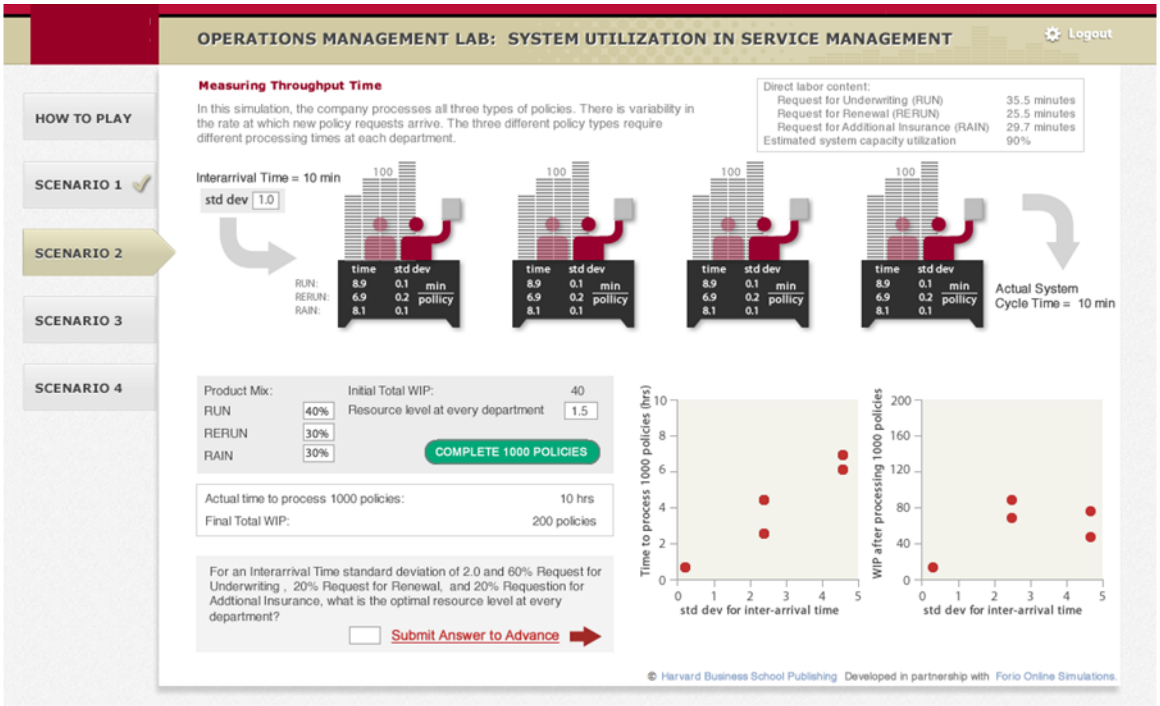

Figure 3: This is an example of a mockup. You can see here the full detail and color present. Full attention is paid to not only layout, but each user interface element

In addition to reviewing the mockups during iteration #3 the team also reviews the initial set of wireframes for the faculty and administration screens. This portion of the simulation typically contain pages with some or all of the following elements:

1. A table showing a list of runs generated by all students in the class. This table will contain the name of the student, the run number, as well as any pertinent inputs and outputs from the run. These metrics are decided upon by the subject matter expert and the modeler
2. Histograms showing student performance across a variety of key metrics. These histograms allow the faculty to quickly and easily determine how their students are performing at the task.
3. A listing of all students in the class, as well as their login ids, and a button to generate any login credentials that the students may request.
4. A page which controls simulation operation. Specifically the ability to open and close the simulation for student interaction, and any specific settings to enable or disable any scenarios which the student may optionally play.
5. A page with introduction videos showing the faculty how to operate and use the simulation, as well as links to any relevant teaching guides and information that they can use to insert the simulation into their curriculum.

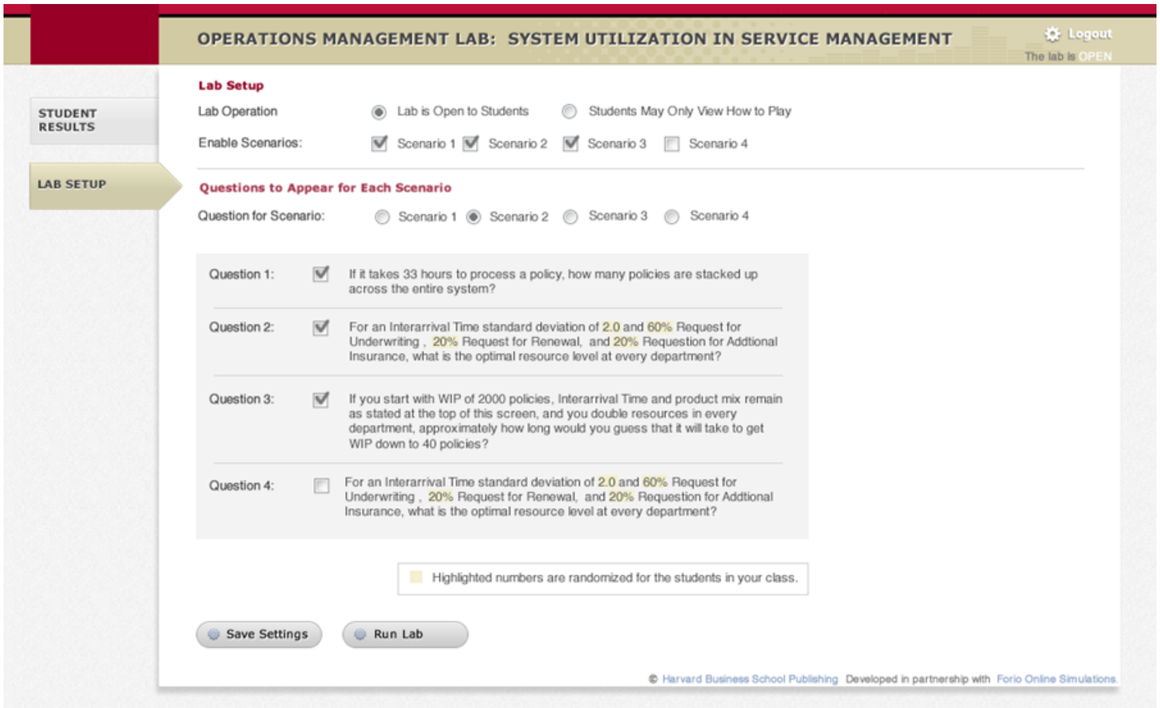

Figure 4: An example of a full color mockup of a faculty setup screen. Notice the options for controlling simulation status etc.

The final portion of iteration #3 is devoted to a review of any updated model behavior, and any impacts that this behavior will have on the learning objectives and the simulation design.

One week after iteration #3 the team again meets for iteration #4 where the student side mockup revisions are reviewed in addition to the first iteration of the faculty mockups. The purpose of this meeting is the last chance for the simulation development group to revise the mockups before they are shown to the wider client audience as a part of paper usability testing.

The next portion of the simulation design phase is set aside for paper usability testing, and the shopping of the design around the client organization. By this point there is enough material that the simulation can be presented to chosen members of the simulation target audience for comment. These commenters are asked to envision themselves playing the simulation, clicking on links and navigating from page to page all on paper. Participants are asked observed looking for any areas where they may become confused, or lost in the page to page navigation, or in the content and backstory of the simulation. Notes are taken as to each stumbling block the participants encounter and are then subsequently reviewed by the team afterwards in preparation for the final design phase iteration.

As the usability testing is occurring the design is also shown to any key client stakeholders who have expressed an interest in the project. This is done so that they may comment on the design before it is signed off on an agreed to be the specification for the development. This helps to foster a feeling of participation from those who would normally not have anytime to participate in the laborious act of simulation design, and it helps to keep in check any unanticipated design decisions made by the simulation design team. Any comments about the mockups from these stakeholders are recorded and addressed for the final iteration.

The fifth and final design phase iteration is set aside to review and incorporate any changes in design from the external team members and usability testers. Comments and feedback are reviewed strictly according to the learning objectives originally developed by the team. After this meeting the final set of mockups and model results are agreed upon as being complete, and the scope for the development project has now been fully explored and specified.

The development phase

The development phase is not as clear cut and generalizable as the simulation design phase. The simulation development phase also follows an iterative development plan, but the number and duration of iterations is determined based on the scope set forth at the end of the design phase. Typically development iterations are longer then design iterations in order to account for the technical coding that must be done which is typically more expensive per hour and takes more hours to accomplish then design.

The first iterations of the development phase are focused on producing a useable prototype of the student side user interface. The typical iteration #1 deliverable contains a way for the subject matter expert to enter in decisions and view key outputs that would appear on a typical simulation dashboard. This prototype does not initially work in all browsers (in the case of web-based simulation development) and implements only a small select portion of mockups produced during the design phase. The reason that iteration #1 has such a relatively simple deliverable is that appropriate time must be allocated to the user interface developer to construct all the code necessary to power the user interface. Typically simulation development is most difficult at first because of the need to setup the systems and APIs (application programming interfaces) that the user interface developer will use throughout the simulation development process.

In some cases where there are multiple similar simulation products being developed the simulation development team will decide to produce a common set of templates that can be re-used across all related simulations. This template represents a way to ease development risk across a variety of projects by abstracting general development processes into a single codebase. Templates when developed correctly can reduce code development on specific projects, make maintenance and support much easier, and reduce product specific iteration cycle times. Templates are not cost free though, they require an upfront investment, and if done poorly, or if they try to generalize too many distinct topics can end up as overly complicated and abstract projects which are hard to debug and re-use. When deciding to do develop a template ample time must be left to develop the template to completion otherwise all the positives from developing a template are not achieved because specific projects will be based on different versions of the template and the project is still forced to deal with the additional costs and potential overhead.

Before each of the development iteration meetings around one-third of the hours invested into development will be set aside for testing. Testing refers to a variety of testing methods from the software development industry such as: manual testing, automated testing, unit testing, regression testing etc. Early on we rely heavily on manual testing, working with quality assurance (QA) testers to develop test plans based on the deliverables set out for each iteration. The reason for the heavy reliance on manual testing is that automated testing platforms generally require a level of stability to the code which is not present early on in the development process. The QA testers work independently of the user interface developer recording bugs, and deviations from the test plan into a bug tracking database which is shared among the testers and the user interface developer. Before each iteration meeting the project manager works with the user interface developer to prioritize issues for completion and to verify the deliverable before sharing it with the client.

About half way through the development process the user interface developer switches focus from the student user interface to the administration and faculty screens. When simulation development has reached a place where the student user interface matches the original mockups and is useable cross browser, and the faculty user interface is in the prototype phase the simulation is beta tested. Simulation beta testing requires the use of a sample target users, which will not be upset when encountering potentially major mistakes, and bugs in simulation output. The users must be experienced enough as users of simulations to be able to comment on simulation usability, design, and level of engagement. These users usually include between seven and twenty participants and are typically paid for two hours of their time to test and comment on the simulation. Comments from the beta testers are weighed against the original learning objectives and appropriate changes to simulation content and design are made as a consequence of this testing.

The next set of user testing is called field testing. This occurs at the point when both the student side user interface and faculty user interface have reach or exceeded parity with the original mockups. Field testing requires finding faculty who are comfortable running the new simulation and potentially encountering minor bugs which do not affect simulation output. Field testers have a much wider level of control over the simulation, and run it without the direct intervention of the simulation development team. This is the first real chance for the simulation to see 'real use'. As always field testers are asked for comment, but at this stage in the project only the most superficial comments be accommodated without major revisions to scope and project duration. Any large scale simulation changes are typically noted and into a separate database for a version 2 of the product. Special attention is paid to field tester comments about the faculty user interface as this is the first time this user interface has been used outside of the simulation development group.

The final iterations of simulation development occur after the field test. At this point a feature freeze is put into effect and no new features are added to the simulation. This is to ensure that the product becomes stable before release. In these final iterations any copy edits, demo development and acceptance testing occurs. As always QA reviews and tests against their test plans, in addition to developing a suite of automated regression tests which can be used during the lifetime of the simulation in order to ensure simulation stability after maintenance changes. The last piece of testing for the simulation is called load testing which is done by simulating 100% above maximum predicted simultaneous users using automated traffic generating tools. This is to ensure that the simulation development team knows how many simultaneous users the system can support and can adjust server side resources as necessary.

Conclusions

We have developed a simulation development methodology that we feel can be broadly applied in the system dynamics field in order to help develop better products which increase the visibility of the field. The development process is by no means complete and is still a work in progress, but we feel that it has come far enough to be of general use to the field. The mathematical model is only a small part of the simulation development effort and much effort is expended not only on expected areas such as user interface development, but also faculty user interface development and testing, whether that be manual, automated, or even beta testing, usability testing and field testing.

References

1. Wendy Grossman, Designed for Life, New Scientist, October 5, 2002, vol. 176, Page 2363
2. Patrick Laughlin, Megan Zander, Erica Knievel, et al., Groups Perform Better Than the Best Individuals on Letters-to-Numbers Problems: Informative Equations and Effective Strategies, Journal of Personality and Social Psychology, 2003, vol. 85(4), Pages 684-694.
3. Craig Larman, Agile and Iterative Development: A Manager's Guide. Addison-Wesley Professional, 2004
4. William Lidwell, Kritina Holden, Jill Butler, Universal Principles of Design, 2003, Page 74.
5. Chow Tsun, Cao Dac-Buu, A survey study of critical success factors in agile software projects, Journal of Systems and Software, Volume 81, Issue 6, June 2008, Pages 961-971