

GSD Sim - A Global Software Development Game



Software development is often characterized as a “wicked problem” due to changing requirements and the realization that the problem to be solved is not really understood until a solution is created.

Global software development (GSD) introduces a host of additional complexities to software development as a result of barriers such as geographic separation, timezone differences, and language and cultural gaps. These barriers, collectively called “global distance,” prevent the kind of informal encounters that co-located teams use to communicate.

A common approach to teaching software engineering concepts is to have students form teams to create a software product; this allows them to experience the problems first hand. However, this approach is much more difficult for GSD, due to the need to have distributed project teams, which come with all the problems caused by global distance that affect real developers.

We developed a *serious game* called **GSD Sim**, that casts players in the role of project manager for a globally distributed software project. Players allocate teams of programmers to different locations around the world, and assign these teams to develop modules that comprise the software product.

The goal of the game is to maximize profit, by reducing costs or increasing revenue. So, a player might outsource development to a low-cost location in effort to save money. Another player might hire high-productivity developers in a high-cost location, in order to finish the project ahead of schedule and thereby realize additional revenue.

The final score is revenue generated during the first six months of sales, less costs incurred:

$$Score = [Sales_{month} \times 6] - \sum_{loc=1}^n (DevCost_{loc} \times NumDev_{loc})$$

The player is given a budget to hire developers. If the development costs exceed the budget, a 25% penalty is applied to costs over the budget limit. Also, if the project is delivered late, the total of sales are reduced by each month past the deadline. However, if the project is completed early, an additional revenue months are added.

Having allocated developers and modules to locations, the simulation begins. The player will have to cope with problems that crop up along the way, such as:

- » a site falls more than 25% behind schedule on a task;
- » a module fails to integrate properly;
- » a module fails system tests;
- » a module fails to deploy correctly;
- » a module or product fails acceptance tests (i.e., fails to meet real requirements);
- » a module fails to deploy correctly.

Problems are more likely to occur at locations that have greater “Global Distance” from the home site, where Global Distance is a combination of geographic distance, timezone overlap, and cultural differences:

Qualitative measure of Global Distance	Impact	Value
Same region (two hour drive)	Low	1
Less than three hour flight	Medium low	2
Transcontinental flight	Medium high	3
Intercontinental flight	High	4
Five hour timezone overlap	Low	1
Three to five hour timezone overlap	Medium low	2
Less than three hour timezone overlap	Medium high	3
No timezone overlap	High	4
Lack of a common language	High	4
Uneven common language skills	Medium high	3
East/West cultural divide	Medium high	3
Mixed High and Low context cultures	Medium high	3
Different national cultures	Medium Low	2
Different organizational cultures	Low	1

Table 1. Global Distance factors and impacts.

The player can implement tactical interventions to cope with these problems:

Tactical Intervention	Cost
Replace manager who quits	15,000
Buy pizza to improve morale	300
Raise salaries to improve morale	2,000
Encourage overtime to recover schedule slip	10,000

Table 2. Tactical Interventions to address immediate problems.

Some of these may improve or reduce morale and therefore productivity.

The player can also attempt to *prevent* problems by making certain strategic interventions, which reduce the probability of an adverse event occurring by reducing global distance.

Intervention that reduces Global Distance	Impact	Cost
Face-to-face meetings	High	25,000
Exchange program	High	125,000
Synchronous communication possibilities	Medium high	5,000
Support for video conference at all sites	Medium low	5,000
Suitable selection of communication tools	Medium low	5,000
Cultural Training	Medium high	25,000
Cultural Liaison/Ambassador	Medium high	125,000
Adopt low-context communication style	Medium Low	5,000
Reduce interaction between teams from different cultures	Low	5,000

Table 3. Strategic Interventions to reduce Global Distance

GSD Sim allows students to experience the difficulties involved in GSD from a project manager's point of view, in a much shorter time and at lower cost than a real global software development project. The game is also entertaining in its own right.

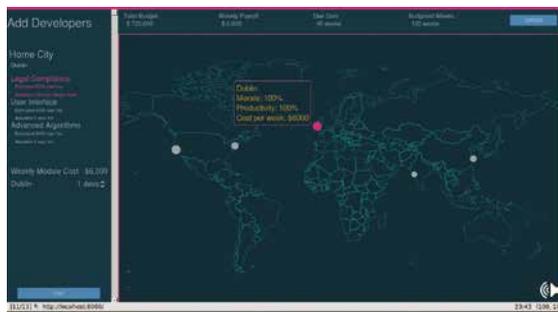


Fig 1. Planning phase: allocating developers to modules and locations

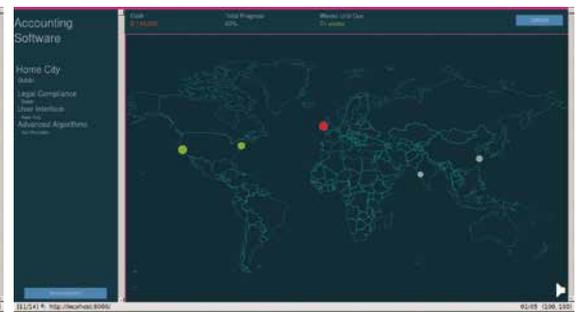


Fig 2. Project progress: home site behind schedule (red), other sites on-schedule (green)

GSD Sim has the potential to reinforce conceptual instruction with hands-on learning at a fraction of the time and cost that would be required to run a true distributed software project.



Fig 3. Status inquiry

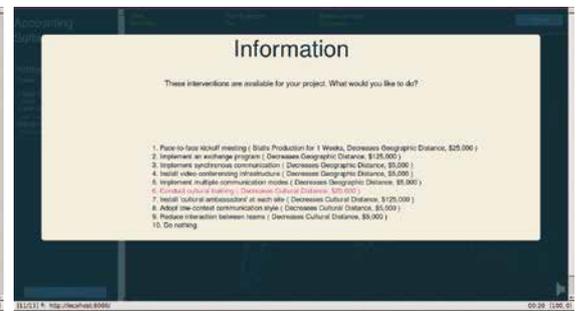


Fig 4. Strategic interventions

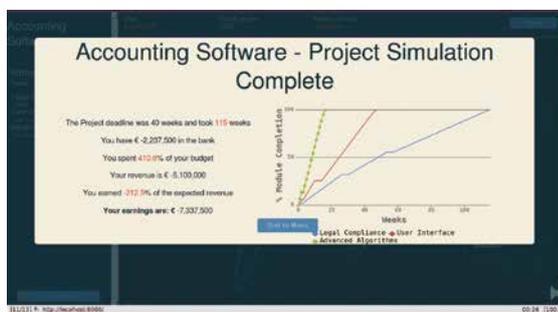


Fig 5. Failed project: over budget, late.

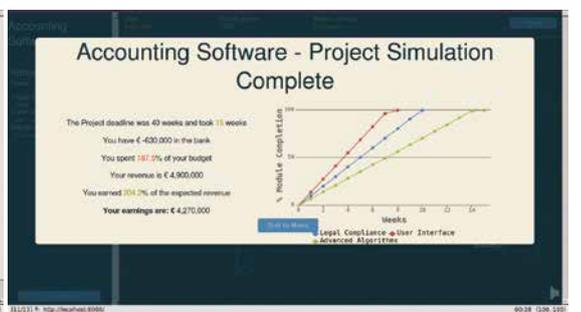


Fig 6. Successful project: over budget, but exceeded revenue projections