

Secret Ninja Testing with HALO¹ Software Engineering

Jonathan Bell, Swapneel Sheth, Gail Kaiser

Department of Computer Science, Columbia University

New York, NY 10027

{jbell, swapneel, kaiser}@cs.columbia.edu

1. There is no connection between our proposal and any games with similar names. The authors have no affiliations with any companies involved in producing such games... except as customers.

Motivation

- Early exposure to testing can be beneficial
- Testing often is not included in early CS curricula
- Testing can be tedious - there is low student engagement and interest in testing

Gamification

- We propose a social approach to expose students to software testing using *gamification*
- Our gamification portrays tasks as *quests* within a storyline
- Quest structure helps make work *satisfying*
 - Clearly lay out goals
 - Shows why the goal matters
 - Shows how to get to the goal, step by step
 - Shows a certificate that will prove success

HALO Game Mechanics

- HALO (Highly Addictive, socialLly Optimized SE) is our gamification platform
- Eclipse plugin
- Maps various SE features to game world
 - Doesn't have to follow the MMORPG concepts strictly - concepts from other games or familiar to conventional SE like priorities and deadlines can be added
 - “Parties” and “guilds” become analogs for teams

HALO Game Mechanics (2)

- Quests can be created in HALO during the initial planning stages for SE projects
- New quests and quest series can be added on the fly, as needed
- Focuses on *social rewards*

Social Rewards in HALO

- Harness *operant conditioning* and peer recognition
 - Grant users “titles” that are displayed throughout the game
 - Allow users to progress in a “level” system
- Global Leader Boards display student progress towards completing quests and achievements
- Rewards scaled with time in game (becoming less common but more valuable); larger rewards for more difficult tasks and collaborating with others

Social Testing with HALO

- To be used this fall in a Data Structures class at Columbia
- Employs a “secret ninja” method (a la Kiniry and Zimmerman)
 - We create quests that result in testing
 - ... but do not call it “testing”
- Example: Subway card vending machine assignment
 - Students design and implement a system for dispensing subway cards
 - Create a story in HALO in which some character needs to use the subway
 - We create *scenarios* to use the system, i.e., test cases

Sample HALO Quests

- Justice League (a fictional group of comic book heroes) need a temporary hide-out
- Need to go apartment hunting in Gotham City
- To appear as “superheroes of the people,” our heroes decide to take the subway
- Batman tries to use his “BatCard” credit card
- Flash doesn’t have exact change
- Green Lantern needs to refill a card
- Last quest may be a “boss” level, requiring students to collaborate

HALO Testing Interface

Dashboard

Jon Bugslayer Bell

Experience Points: Level 1 | 150 of 200 | Level 2

Achievements: 20 points of 420

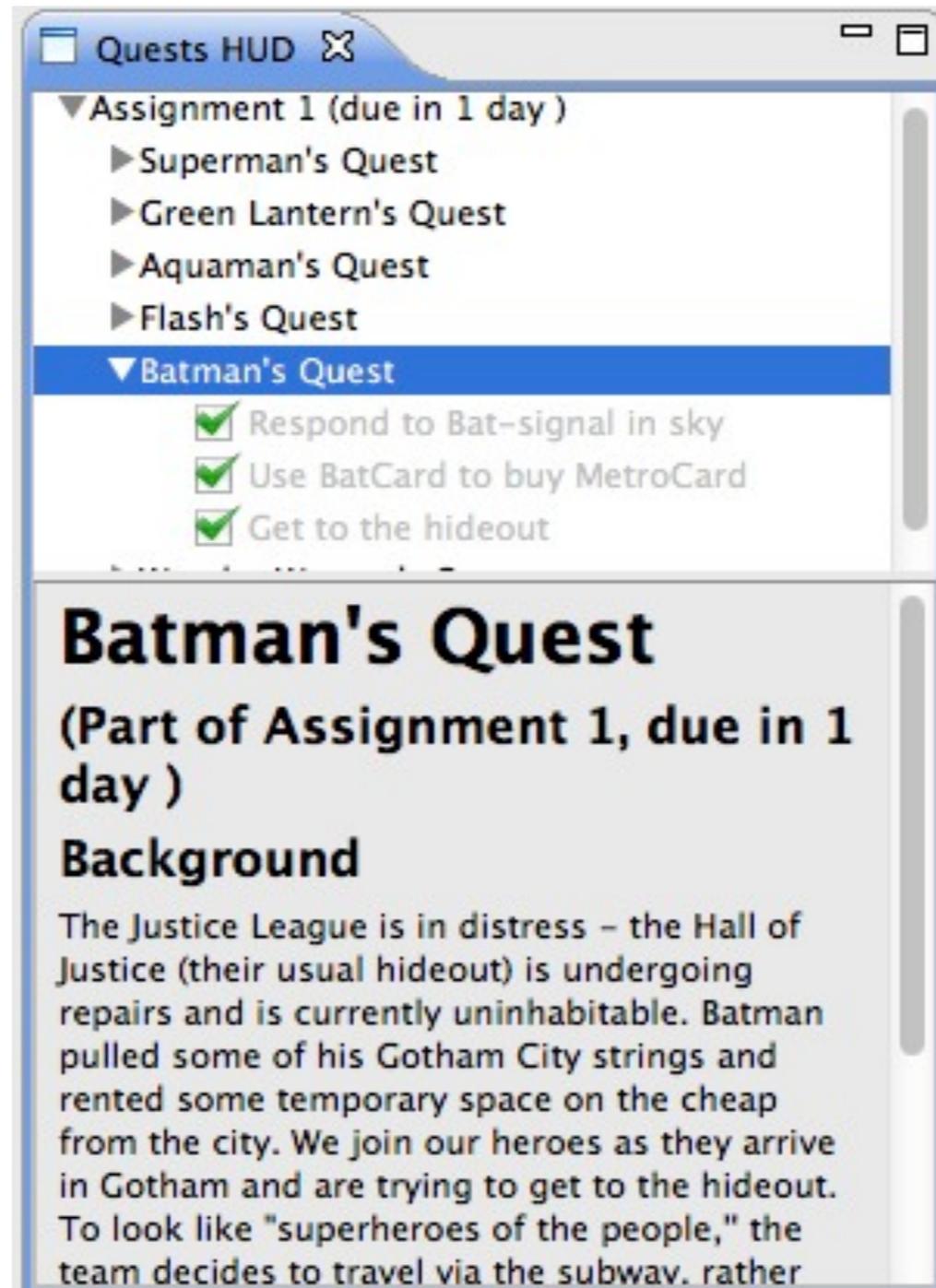
Recent Achievements:
Master of Bugs - Aug 2, 2011

Quest Progress: 1 of 5

Assignment	Progress	Due Date
Assignment 1	1 of 1 quests	Due in 1 day
Assignment 2	0 of 1 quests	Due in 1 week
Assignment 3	0 of 1 quests	Due in 1 week and 3 days
Assignment 4	0 of 1 quests	Due in 2 weeks
Assignment 5	0 of 1 quests	Due in 3 weeks

Change Password | Logout

HALO Testing Interface



HALO Testing Interface

- Currently, heavy on text, hope to work in graphics and avatars a la “clippy”
- Paired with a community site
- Instructor portal for reporting metrics

Related Work

- Regarding testing in CS education
 - Integrate testing across all CS courses - Jones [12, 13]
 - “Test-first” SE curriculum - Edwards [5]
 - Students provide test cases that are used on others’ code - Goldwasser [8]
- Regarding “secret ninja” formal methods - Kiniry and Zimmerman [14]
- SE Games - Software Hut [10], SimSE[17], etc.

Research Agenda

- How do we appeal to students and improve their approach to the SDLC?
- How can we ensure that “everybody wins”?
- What requirements must we make sure we fulfill to appeal to instructors?
- Can we design quest templates that would work for many different classes?
- What other areas can HALO be applied to?

Conclusion

- New approach for teaching students software testing using HALO (Highly Addictive, socialLly Optimized) SE
- We believe that our Secret Ninja Software Testing approach will make testing more engaging and fun for students
- We feel that this will inculcate good software testing habits at an early stage

Acknowledgements

- The authors are members of the Programming Systems Laboratory, funded in part by NSF CNS-0717544, CNS-0627473 and CNS-0426623, and NIH 2 U54 CA121852-06

enable (v_t) : to make possible, practical, or easy



PROGRAMMING SYSTEMS LAB
COLUMBIA UNIVERSITY



<http://www.psl.cs.columbia.edu/>

Secret Ninja Testing with HALO Software Engineering

Jonathan Bell, **Swapneel Sheth**, Gail Kaiser

Department of Computer Science, Columbia University

New York, NY 10027

{jbell, swapneel, kaiser}@cs.columbia.edu

Student Software Testing

- Conducted a brief study using student assignments from Columbia's *Object Oriented Programming and Design in Java* class
 - Assignments are not “hard” to program - they are exercises in design
- Many students (33%) had at least one “major” functionality flaw
- Most students (85%) had at least one “minor” functionality flaw
- Suggests much room to improve with testing