Center for Teaching and Learning

Technical Report

Towards Model Driven Game Engineering in SimSYS:

Requirements for the Agile Software Development Process Game

by

Kendra M. L. Cooper, The University of Texas at Dallas
C. Shaun Longstreet, Marquette University

Working Paper MU-CTL-01-12

Towards Model Driven Game Engineering in SimSYS: Requirements for the Agile Software Development Process Game

UTDCS-06-12
The University of Texas at Dallas

MU-CTL-01-12
Marquette University

March 2012

Kendra M. L. Cooper, The University of Texas at Dallas
C. Shaun Longstreet, Marquette University

Abstract

Software Engineering (SE) and Systems Engineering (Sys) are knowledge intensive, specialized, rapidly changing disciplines; their educational infrastructure faces significant challenges including the need to rapidly, widely, and cost effectively introduce new or revised course material; encourage the broad participation of students; address changing student motivations and attitudes; support undergraduate, graduate and lifelong learning; and incorporate the skills needed by industry. Games have a reputation for being fun and engaging; more importantly immersive, requiring deep thinking and complex problem solving. We believe educational games are essential in the next generation of e-learning tools. An extensible, freely available, engaging, problem-based game platform that provides students with an interactive simulated experience closely resembling the activities performed in a (real) industry development project would transform the SE/Sys education infrastructure.

Our goal is to extend the state-of-the-art research in SE/Sys education by investigating a game development platform (GDP) from an interdisciplinary perspective (education, game research, and software/systems engineering). A meta-model has been proposed to provide a rigourous foundation that integrates the three disciplines. The GDP is intended to support the semi-automated development of collections of scripted games and their execution, where each game embodies a specific set of learning objectives. The games are scripted using a template based approach. The templates integrate three approaches: use cases; storyboards; and state machines (timed, concurrent, hierarchical state machines). The specification templates capture the structure of the game (Game, Acts, Scenes, Screens, Challenges), storyline, characters (player, non-player, external), graphics, music/sound effects, rules, and so on. The instantiated templates are (manually) transformed into XML game scripts that can be loaded into the SimSYS Game Play Engine. As a game is played, the game play events are logged; they are analyzed to automatically assess a player's accomplishments and automatically adapt the game play script.

Currently, we are manually defining a collection of games. The games are being used to ensure the GDP is flexible and reliable (i.e., the prototype can load and correctly run a variety of game scripts), the ontology is comprehensive, and the templates assist in defining well-organized, modular game scripts. In this report, we present the initial part of an Agile Software Development Process game (Act I, Scenes 1 and 2) that embodies learning objectives related to SE fundamentals (requirements, architecture, testing, process); planning with Gantt charts; working with budgets; and selecting a team for an agile development project. A student player is rewarded in the game by getting hired, scoring points, or getting promoted to lead a project. The game has a variety of settings including a classroom, job fair, and a work environment with meeting rooms, cubicles, and a water cooler station. The main non-player characters include a teacher, boss, and an evil peer.

In the future, semi-automated support for creating new game scripts will be explored using a wizard interface. The templates will be formally defined, supporting automated transformation into XML game scripts that can be loaded into the SimSYS Game Engine. We also plan to explore transforming the requirements into a notation that can be imported into a commercial tool that supports Statechart simulation.

Table of Contents

List of Figures

List of Tables

1. Introduction

Software Engineering (SE) and Systems Engineering (Sys) are knowledge intensive, specialized, rapidly changing disciplines; their educational infrastructure faces significant challenges including the need to rapidly, widely, and cost effectively introduce new or revised course material; encourage the broad participation of students; address changing student motivations and attitudes; support undergraduate, graduate and lifelong learning; and incorporate the skills needed by industry. Games have a reputation for being fun and engaging; more importantly immersive, requiring deep thinking and complex problem solving. We believe educational games are essential in the next generation of e-learning tools. An extensible, freely available, engaging, problem-based game platform that provides students with an interactive simulated experience closely resembling the activities performed in a (real) industry development project would transform the SE/Sys education infrastructure.

Our goal is to extend the state-of-the-art research in SE/Sys education by investigating a game development platform (GDP) from an interdisciplinary perspective (education, game research, and software/systems engineering). A domain meta-model has been proposed to provide a rigourous foundation that integrates the three disciplines. The GDP is intended to support the semi-automated development of collections of scripted games, where each game embodies a specific set of learning objectives that are traced to a body of knowledge. In our game engineering methodology, the game is specified using a new set of templates; the specification captures the structure of the game (Game, Acts, Scenes, Screens), storyline, characters (player, non-player, external), graphics, music/sound effects, rules, and so on. As a game is played, the game play events are logged; they are analyzed to automatically assess a player's accomplishments and automatically adapt the game play script. New templates are proposed in this work for capturing serious game requirements: Game; Act; Scene; Screen; and a Quiz challenge. The instantiated templates are (manually) transformed into XML game scripts that can be loaded into the SimSYS Game Play Engine. The game requirements define a timed, hierarchical, concurrent state machine.

Currently, we are manually defining a collection of games. The games are being used to ensure the GDP is flexible and reliable (i.e., the prototype can load and correctly run a variety of game scripts), the domain meta-model is comprehensive, and the templates assist in defining well-organized, modular game scripts that effectively communicate the game requirements. In this report, we present:

- the initial part of an Agile Software Development Process game (Act I, Scenes 1 and 2) that embodies learning objectives related to SE fundamentals (requirements, architecture, testing, process); planning with Gantt charts; working with budgets; and selecting a team for an agile development project. A student player is rewarded in the game by getting hired, scoring points, or getting promoted to lead a project. The game has a variety of settings including a classroom, job fair, and a work environment with meeting rooms, cubicles, and a water cooler station. The main non-player characters include a teacher, boss, and an evil peer.
- description of the SimSYS domain meta-model (Appendix B).
- descriptions for the SIMSYS template definitions for organizing a game into Acts, Scenes, Screens and Challenges (Appendix C). The templates are informally defined in this report: they are tables described using natural language. When additional templates are needed for different kinds of challenges, they can be readily added using this modular approach; in addition making changes to the templates has been straightforward.

This report is organized into the following main sections. Section 2 provides an overview of the Agile Software Development Process Game; Section 3 provides the detailed requirements

specification for the first part of the game (Act I, Scenes 1 and 2). The Conclusions and Future work are discussed in Section 4. A set of Appendices are included to present the Employee Character Profiles (Appendix A), SimSYS Foundations (Appendix B), and the Game Templates (Appendix C).

2. Game Overview

2.1 Organization of the Game

A simplified UML Use Case diagram provides a visual overview of the structure of the example game in this report (Figure 1). A summary of the game is available in Section 2; the detailed description is available in Section 3. In the use case diagram, use cases representing a Game are defined with a Game template, use cases representing an Act are defined with an Act template and so on. We found that defining a template for each level of abstraction in the game provided a clear, easy to understand specification, as opposed to defining one general purpose template that could be tailored for different levels (perhaps by leaving some part blank or marking them not applicable). The templates defined to support the game specification are available in Appendix C.

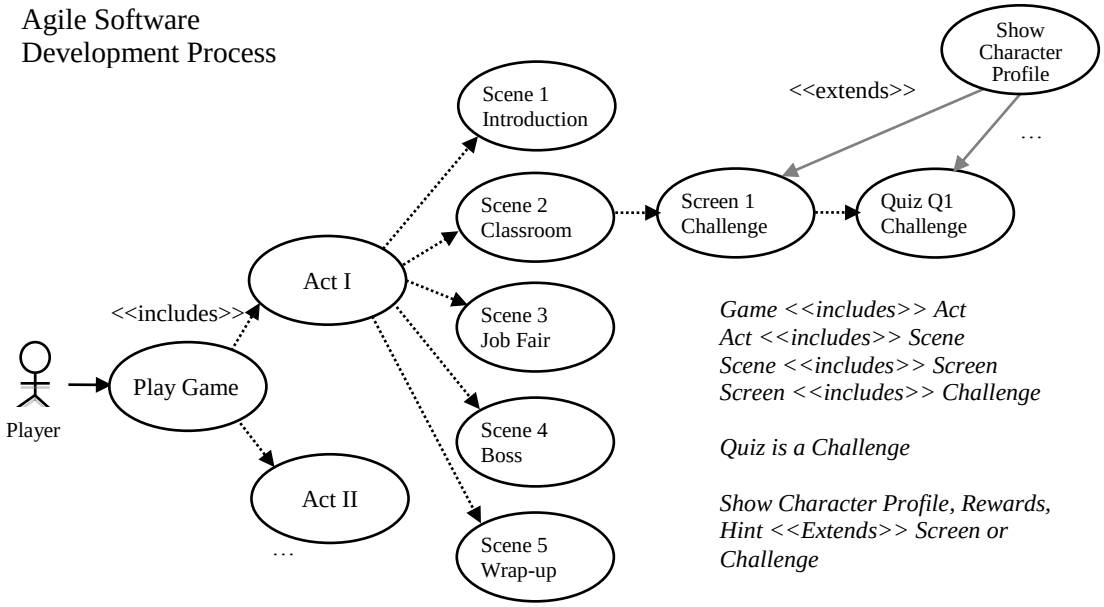


Figure 1. Agile Software Development Process Game: A Simplified UML Use Case Diagram

2.2 Characters

SimSYS characters in this game include the Player and non-player (main, employee). Each character has a profile that is organized like a resume (Appendix A). There are no external characters in this game.

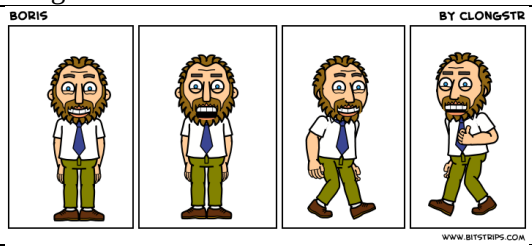
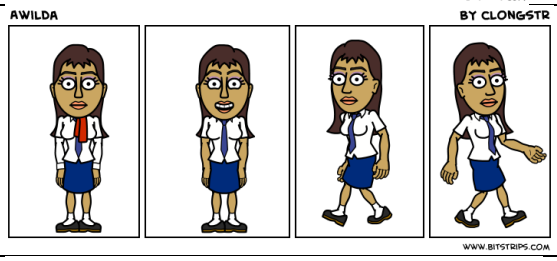
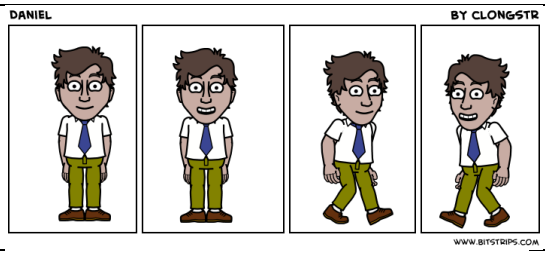
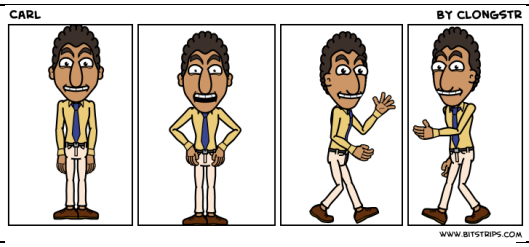
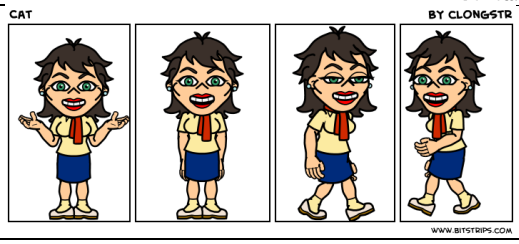
2.2.1 Player Character

The player is the protagonist in this game; he/she is a software engineering student who progresses through the game challenges to acquire rewards (including getting hired, points, certificates, awards, promotions) or receive penalties (getting demoted or fired, point penalties, revocation of awards or certificates) by taking on challenges in the game. At the beginning of the game, the player chooses one of the six options (Table 1); they name their character.

2.2.1.1 Player Character Avatar Options

There are six player avatar options in the game. In this specification Player 3 is used to illustrate the Player character.

Table 1. Character Avatar Options

Character	Image
Player 1	
Player 2	
Player 3	
Player 4	
Player 5	
Player 6	

2.2.1.2 Player Character Profile

Name	TBD
Resume Photo	
Title	Software Engineering Student
Skills	Software engineering, Unified Process, agile methods, project management, OO programming, IDE, configuration management, UML case tool, UM notation.
Years of Experience	0
Communication	Very good
Very good	Very good
Teamwork	Very good
Demographics	Male, Caucasian
Availability	
Attendance	
Degrees	None B.Sc. senior, Steven B. Allmer School of Software Engineering

2.2.2 Player Environment

2.2.2.1 Main Non-player Characters

The main non-player characters in the game are (Table 2):

Table 2. Main Non-player Characters

Character	Description/Role	Ontology
UR Boss	The Player's supervisor at work.	Director
Dr. Ima Coder	The Player's software engineering instructor at school.	Interlocutor
Nim Esis	CEO's nephew and the Player's peer at school. Competes with the Player at school and in the workplace. He tries to hamper the Player's success and take credit for the Player's accomplishments.	Antagonist

2.2.2.1.1 Profiles

The main non-player character profiles are illustrated here (Table 3,Table 4, Table 5).

Table 3. UR Boss Profile

Name	Ur Boss
Resume Photo	
Title	Project Manager
Skills	**Project Management**
Years of Experience	7
Communication	Excellent
Leadership	Excellent
Teamwork	Excellent
Demographics	Male, African American
Availability	
Attendance	
Degrees	B.B.A.**, Harvard University** **M.B.A. Harvard University**

Table 4. Dr. Ima Coder Profile

Name	Dr. Ima Coder
Resume Photo	
Title	Instructor
Technical Skills	Software Engineering, Project Management, Capstone Project courses
Years of Experience	5
Communication	Excellent
Leadership	Excellent
Teamwork	Excellent
Demographics	Female, Caucasian
Availability	
Attendance	
Degrees	B.Sc., Computer Science, Stanford University Ph.D. Computer Science, Stanford University

Table 5. Nim Esis Profile

Name	Nim Esis
Resume Photo	
Title	Student
Technical Skills	Java programming language, C# programming language, OOAD, Software Engineering
Years of Experience	0
Communication	Very good
Leadership	Very good
Teamwork	Very good
Demographics	Male, Caucasian
Availability	
Attendance	
Degrees	None B.Sc., senior, Steven B. Allmer School of Software Engineering

2.2.2.2 Company Employees

The company employee characters in the SimSYS game are (
Table 6):

Table 6. Company Employee Characters

Character Name	Title	Role	Meta-Model type
Aadrika Baker	Junior software designer	Co-Worker	Constructor
Jane Baker	Junior software designer	Co-Worker	Constructor
Berg Barker	Junior requirements engineer	Co-Worker	Constructor
Alpino Carter	Junior software designer	Co-Worker	Constructor
Montane Chandler	Intermediate manager	Co-Worker	Constructor
Rahul Cook	Intermediate software developer	Co-Worker	Constructor
Li Cooper	Intermediate software designer	Co-Worker	Constructor
Eithne Fletcher	Intermediate software requirements engineer	Co-Worker	Constructor
Kilimo Hansard	Junior software developer	Co-Worker	Constructor
Thomas Miller	Senior software architect	Co-Worker	Constructor
Sierra Proctor	Junior requirements engineer	Co-Worker	Constructor
Ann Smith	Senior software developer	Co-Worker	Constructor
Gora Stone	Intermediate Business analyst	Co-Worker	Constructor
Marco Thatcher	Senior software testing	Co-Worker	Constructor
Luo Tyler	Intermediate software tester	Co-Worker	Constructor
Capri Ward	Senior requirements engineer	Co-Worker	Constructor
Bob Weaver	Junior software tester	Co-Worker	Constructor

2.2.2.2.1 Sample Profile

A sample profile is illustrated here (Table 7). The set of employee profiles is in Appendix A.

Table 7. Sample Employee Profile

Name	Berg Baker
Resume Photo	
Title	Junior requirements engineer
Technical Skills	Petri nets, statecharts, UML use cases, IEEE 830
Years of Experience	4 requirements
Communication	Great
Leadership	Fair
Teamwork	Great
Demographics	Caucasian American, Female
Availability	M-F, 8 a.m. – 5 p.m.
Attendance	95%
Degrees	B.Sc. Computer Science, NorthEastern University

2.2.2.3 External Characters (vendors, customers, business organization)

None.

2.3 Definitions

2.3.1 Timers

Transition timing terminology:
SLOW TRANSITION= 5 seconds
MODERATE TRANSITION = 3 seconds
QUICK TRANSITION = 1 seconds

Presentation duration timing terminology:
SLOW PRESENTATION= 5 seconds
MODERATE PRESENTATION = 3 seconds
QUICK PRESENTATION= 1 seconds

Hint timer terminology:
SLOW HINT TIMER = 5 seconds
MODERATE HINT TIME = 3 seconds
QUICK HINT TIMER = 1 seconds

Animation movement timing terminology:
SLOW MOVEMENT= 5 seconds
MODERATE MOVEMENT = 3 seconds
QUICK MOVEMENT = 1 seconds

Animation effect timing terminology:
SLOW EFFECT= 5 seconds
MODERATE EFFECT = 3 seconds
QUICK EFFECT = 1 seconds

2.3.2 Style

The BitStrips style is used to define the look ‘n feel of the GUI.

2.3.3 Character Interactions (Emotional Responses)

Table 8. Character Interactions (Emotional Responses)

	Response				
	Protagonist (player)	Antagonist (Nim Esis)	Boss (Ur Boss)	Educator (Dr. Ima Coder)	Colleagues
Protagonist (player)					
Rewarded	Very Happy	Very Unhappy	Happy	Happy	Happy
Punished	Unhappy	Happy	Unhappy	Unhappy	Unhappy
Antagonist (Nim Esis)					
Rewarded	Happy	Very happy	Happy	Happy	Unhappy
Punished	Neutral	Unhappy	Unhappy	Unhappy	Happy
Boss (Ur Boss)					
Rewarded	Happy	Happy	Very happy	Happy	Neutral
Punished	Unhappy	Unhappy	Unhappy	Unhappy	Neutral
Educator (Dr. Ima Coder)					
Rewarded	Happy	Happy	Happy	Very Happy	Neutral
Punished	Unhappy	Unhappy	Unhappy	Unhappy	Neutral
Colleagues					
Rewarded	Happy	Unhappy	Happy	Happy	Very Happy
Punished	Happy	Happy	Unhappy	Unhappy	Unhappy

3. Agile Software Development Process Game Requirements

3.1 Game Description

Note. Learning objectives are defined with respect to the SWEBOK[1].

Table 9. Agile Software Development Process Game

Identifier	Game1
Purpose	This SimSYS game has one Act; which presents challenges on the fundamentals of SE (a quiz), building part of a Gantt chart (fill in an empty chart), and selecting a team for a project using an agile software development process. Note. Act II will be added in the future to proceed with agile development of a project with the team the player selects in Act I.
Uses Acts	Act 1
Learning objectives	Challenges the student on her/his general, background knowledge of: Software Testing Challenge = Interactive Quiz SWEBOK = Software Testing Software Testing Fundamentals Test Techniques Bloom = Knowledge, Comprehension Software Design Challenge = Interactive Quiz SWEBOK = Software Design General Design Concepts Context of Software Design Software Design Process Bloom = Knowledge, Comprehension

[1] Díaz-Herrera J. and Hilburn, T. (editors), Software Engineering 2004 Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering, A Volume of the Computing Curricula Series, August 23, 2004, The Joint Task Force on Computing Curricula, IEEE Computer Society and the Association for Computing Machinery.

<table>
<tr><td></td><td>Software Engineering Process
Challenge = Interactive Quiz
SWEBOK = Software Engineering Process
Process Definition
Bloom = Knowledge, Comprehension

If the player is successful in answering, she/he will be able to move forward in the game and collect points towards a future reward.</td></tr>
<tr><td>Style</td><td>Bitstrips cartoon</td></tr>
<tr><td>Characters</td><td></td></tr>
<tr><td>Player</td><td>Player Character: name is TBD
Meta-model type: Protagonist
Profile options:

<table>
<tr><td>Name</td><td>Daniel</td></tr>
<tr><td>Resume Photo</td><td>  </td></tr>
<tr><td>Title</td><td>Software Engineering Student</td></tr>
<tr><td>Skills</td><td>Software engineering, Unified Process, agile methods, project management, OO programming, IDE, configuration management, UML case tool, UM notation.</td></tr>
<tr><td>Years of Experience</td><td>0</td></tr>
<tr><td>Communication</td><td>Good</td></tr>
<tr><td>Leadership</td><td>Good</td></tr>
<tr><td>Teamwork</td><td>Good</td></tr>
<tr><td>Demographics</td><td>Male, Caucasian</td></tr>
<tr><td>Availability</td><td></td></tr>
<tr><td>Attendance</td><td></td></tr>
</table>
</td></tr>
</table>

<table>
<tr><td></td><td>
<table>
<tr><td>Degrees</td><td>None
B.Sc. senior, Steven B. Allmer School of Software Engineering</td></tr>
</table>

Note. Display Character Profile
Size: LARGE
Location: CS

Rewards
Points: 0
Trophies: none
Certificates: none
Promotion: none
Hint: none
</td></tr>
<tr><td>Non-player</td><td>NPC: Dr. Ima Coder
Meta-model type: Interlocutor
Profile:

<table>
<tr><td>Name</td><td>Ima Coder</td></tr>
<tr><td>Resume Photo</td><td>  </td></tr>
<tr><td>Title</td><td>Instructor</td></tr>
<tr><td>Technical Skills</td><td>Software Engineering, Project Management, Capstone Project courses</td></tr>
<tr><td>Years of Experience</td><td>5</td></tr>
<tr><td>Communication</td><td>Excellent</td></tr>
<tr><td>Leadership</td><td>Excellent</td></tr>
<tr><td>Teamwork</td><td>Excellent</td></tr>
</table>
</td></tr>
</table>

Demographics	Female, Caucasian
Availability	
Attendance	
Degrees	B.Sc., Computer Science, Stanford University Ph.D. Computer Science, Stanford University

Note. Display Character Profile
Size: LARGE
Location: CS

Rewards
Points: 0
Trophies: none
Certificates: none
Promotion: none
Hint: none

NPC: Nim Esis
Meta-model type: Antagonist
Profile:

Name	Nim Esis
Resume Photo	
Title	Student
Technical Skills	Java programming language, C# programming language, OOAD, Software Engineering
Years of Experience	0
Communication	Very good

Leadership	Very good
Teamwork	Very good
Demographics	Male, Caucasian
Availability	
Attendance	
Degrees	None B.Sc., senior, Steven B. Allmer School of Software Engineering

Note. Display Character Profile
Size: LARGE
Location: CS

Rewards
Points: 0
Trophies: none
Certificates: none
Promotion: none
Hint: none

NPC: Ur Boss
Meta-model type: Director
Profile:

Name	Ur Boss
Resume Photo	
Title	Project Manager
Skills	Project Management

<table>
<tr><td></td><td>
<table>
<tr><td>Years of Experience</td><td>7</td></tr>
<tr><td>Communication</td><td>Excellent</td></tr>
<tr><td>Leadership</td><td>Excellent</td></tr>
<tr><td>Teamwork</td><td>Excellent</td></tr>
<tr><td>Demographics</td><td>Male, African American</td></tr>
<tr><td>Availability</td><td></td></tr>
<tr><td>Attendance</td><td></td></tr>
<tr><td>Degrees</td><td>B.B.A., Harvard University
M.B.A. Harvard University</td></tr>
</table>
Note. Display Character Profile
Size: LARGE
Location: CS

Rewards
Points: 0
Trophies: none
Certificates: none
Promotion: none
Hint: none
</td></tr>
<tr><td>Behaviour state machine description:

Current
Transition
event
condition
output
Next</td><td>Initial state for the Game

INITIALIZE Game
Note. The entire game script is loaded.
START Game
PLAY Game
END Game

Final state for the Game</td></tr>
<tr><td>Alternate flow</td><td>If an error occurs, then end the game.</td></tr>
</table>

of events	

3.2 Act 1 Description.

Table 10. Act 1

Identifier	Act 1
Purpose	Act 1 introduces the game, presents challenges on the fundamentals of SE (a quiz), building part of a Gantt chart (fill in an empty chart), and selecting a team for a project using an agile software development process.
Learning objectives	Challenges the student on her/his general, background knowledge of: Software Testing Challenge = Interactive Quiz SWEBOK = Software Testing Software Testing Fundamentals Test Techniques Bloom = Knowledge, Comprehension Software Design Challenge = Interactive Quiz SWEBOK = Software Design General Design Concepts Context of Software Design Software Design Process Bloom = Knowledge, Comprehension Software Engineering Process Challenge = Interactive Quiz SWEBOK = Software Engineering Process Process Definition Bloom = Knowledge, Comprehension If the player is successful in answering, she/he will be able to move forward in the game and collect points towards a future reward.
Uses Scenes	Scene 1 Game Welcome Scene 2 Classroom Challenge Scene 3 Job Fair Challenge (TBD) Scene 4 Boss Challenge (TBD)

	Scene 5 Wrap-up Screen
Behaviour state machine description: Current Transition event condition output Next	Initial state for Act I START Act 1 Play Act 1 End Act 1 Final state for Act I
Alternate flow of events	If an error occurs, then end the game.

3.2.1 Act 1, Scene 1 Introduction

Table 11. Scene 1 - Introduction

Identifier	Scene 1
Purpose	Introduce the look ‘n feel of the game. Provide credit (UTD, developers). Introduce the purpose of the game. Player chooses and names their avatar Introduce the main non-player characters Dr. Ima Coder and Nim Esis.
Learning objectives	Not present.
Uses Screens	Screen1 Welcome Screen 2 Dr. Ima Coder Screen 3 Choose and Name Character Screen 4 Nim Esis
Behaviour state machine description:	Initial state for Scene 1 Start Scene 1 Play Scene 1

Current Transition event condition output Next	End Scene 1 Final state for Scene 1
Alternate flow of events	If an error occurs, then end the game.

3.2.1.1 Act 1, Scene 1, Screen 1: Welcome

Table 12. Game Welcome

Identifier	Screen 1	
Purpose	Present a bright splash page to catch the attention of the player and the Bitstrips user interface look, present the credit page for UTD, and introduce the purpose of the game to the player.	
Learning Objectives	Not present.	
Declarations, initialization		
Challenge	Not present.	
Characters		
Player	Not present	
Non-player characters	Not present	
Setting (visual)		
Backdrop	Cloudscape + Cityscape **Notes.** **The Cloudscape backdrop has a blue background**	

<table>
<tr><td></td><td colspan="3">with animated, cloud set decorations.
The Cityscape has a building with a Softicorp sign on it.</td></tr>
<tr><td>Props</td><td colspan="3"></td></tr>
<tr><td>Generic interaction</td><td colspan="3"><table>
<tr>
<td>Information Box 1
Text: Welcome to SimSys
Size: LARGE
Location: CSL

Information Box 2
Text: BOLD The University of Texas at Dallas
Text: Agile Methods in Software Engineering Game
Size: LARGE
Location: CSL

Information Box 3
Text: Welcome to SimSys! In this game, you will be able to test your skills as a future software engineer development manager. You will be challenged on your basic knowledge of software engineering concepts, but just as important, you will get the chance to strengthen your abilities in balancing time, budget and the complexities of team management in an agile software development process.
Size: LARGE
Location: CS</td>
<td>Right Arrow Button 1
Location: DSL</td>
<td>Hint Thought Bubble 1
Text: Click arrow to continue
Speaker: Arrow Button
Size: SMALL
Location: CSL</td>
</tr>
</table></td></tr>
</table>

<table>
<tr><td>Set Decorations</td><td colspan="2"><table>
<tr>
<td>Cloud 1
Size: MEDIUM
Location: OSR

Animation movement 1
Movement: GLIDE
Timing: SLOW
Loop: YES
Move to: OSL
Path: STRAIGHT

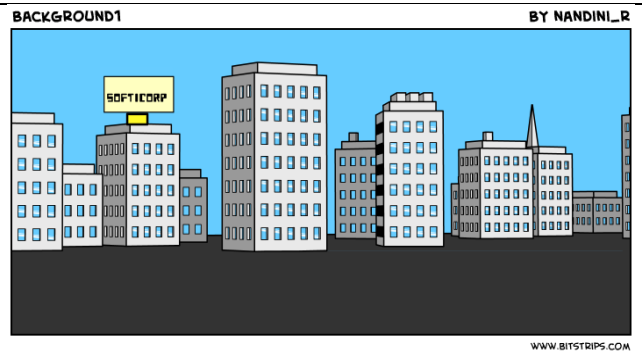
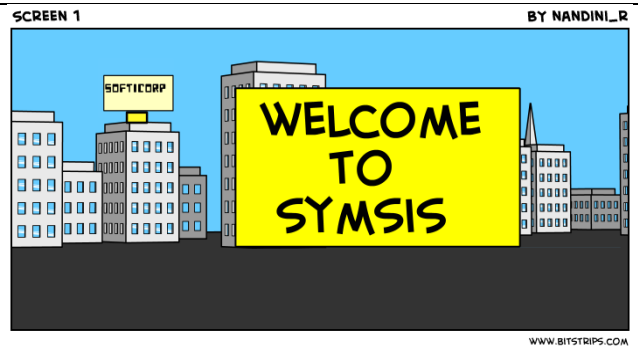
Note. Cloud movement is intended to be on the Cloudscape, behind the Cityscape buildings.</td>
<td>Cloud 2
Size: SMALL
Location: OSR

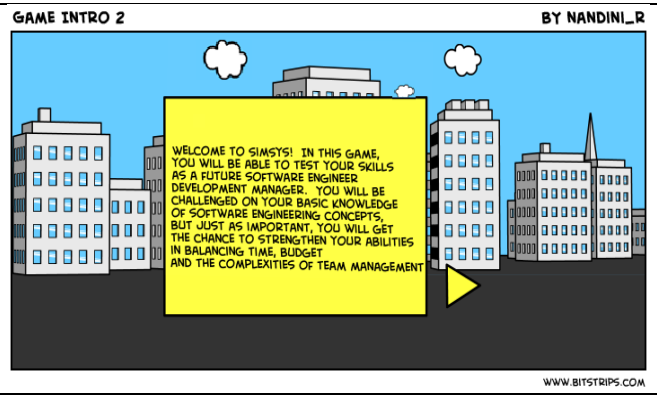
Animation movement 1
Movement: GLIDE
Timing: SLOW
Loop: YES
Move to: OSL
Path: STRAIGHT

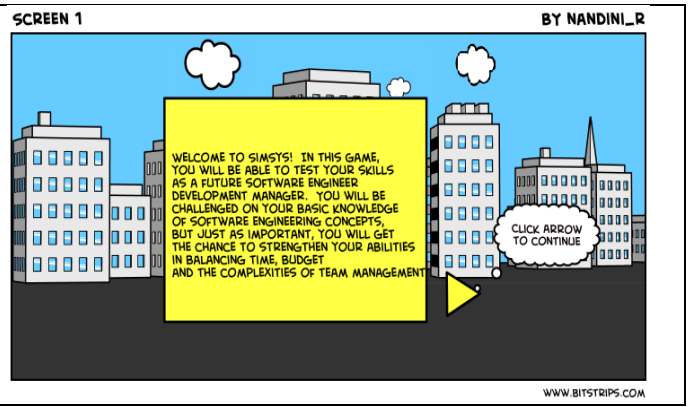
Note. Cloud movement is intended to be on the Cloudscape, behind the Cityscape buildings.</td>
<td>Cloud 3
Size: MEDIUM
Location: OSR

Animation movement 1
Movement: GLIDE
Timing: SLOW
Loop: YES
Move to: OSL
Path: STRAIGHT

Note. Cloud movement is intended to be on the Cloudscape, behind the Cityscape buildings.</td>
</tr>
</table></td></tr>
<tr><td colspan="3">Audio</td></tr>
<tr><td>Music</td><td>Background music</td><td> 
Opening Credits.mid</td></tr>
<tr><td>Challenge</td><td>Not present</td><td></td></tr>
<tr><td colspan="3">Game Play</td></tr>
</table>

Start of Screen	1) Backdrop Cloudscape 2) Backdrop Cityscape with Softicorp sign	
Interactions (normal flow of events)	**Note. Start the background music for the screen.** START Background music **Note. A sequence of information boxes are displayed to the player. The clouds are intended to slowly float by in the background.** START Cloud 1 Animation movement 1 FADE IN the Information Box 1 as a SLOW EFFECT Display for a MODERATE PRESENTATION amount of time. FADE OUT the Information Box 1 as a SLOW EFFECT START Cloud 2 Animation movement 1 FADE IN the Information Box 2 as a SLOW EFFECT Display for a MODERATE PRESENTATION amount of time. FADE OUT the Information Box 2 as a SLOW EFFECT START Cloud 3 Animation movement 1 FADE IN the Information Box 3 as a SLOW EFFECT Display for a MODERATE PRESENTATION amount of time. **Note. When the last information box has been displayed for its presentation time then the right arrow is displayed, which**	

	allows the player to progress in the game. FADE IN the Right Arrow Button 1 as a SLOW EFFECT If the player has not selected the arrow after HINT TIMER MODERATE amount of time, then display the Hint Thought Bubble 1. Player clicks the Right Arrow Button 1 to end the screen. **Note. At the end of the screen, the game needs to stop the audio and remove the visual setting from the display.**	
Alternate flow of events	If an error occurs, then end the game.	
End of Screen	Two options: Hint thought bubble not displayed Hint thought bubble displayed	

3.2.1.2 Act 1, Scene 1, Screen 2: Dr. Ima Coder

Table 13. Dr. Ima Coder

Identifier	Screen 2 Note. Introduce Dr. Ima Coder character	
Purpose	Introduce Dr. Ima Coder, one of the main non-player characters in the game. Dr. Ima Coder is a teacher at the college. Her character type from the ontology is interlocutor.	
Learning Objectives	Not present.	
Declarations, initialization		
Challenge	Not present.	
Characters		
Player	Not present	
Non-player characters	NPC 1: Dr. Ima Coder Pose: STANDING Size: MEDIUM	

<table>
<tr><td></td><td>Location: OSR

Animation movement 1
Movement: GLIDE
Timing: SLOW
Loop: NO
Move to: DSR
Path: STRAIGHT

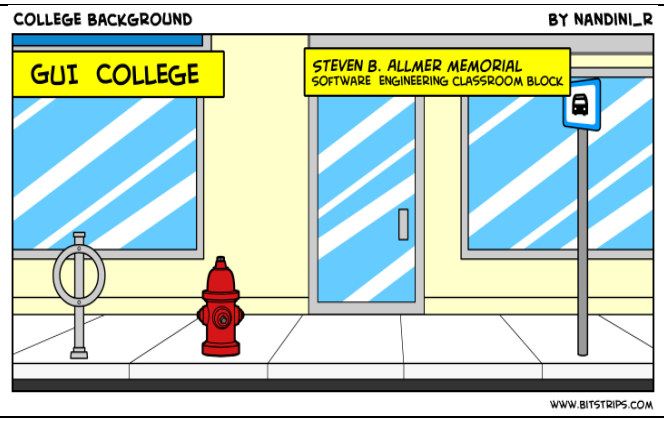
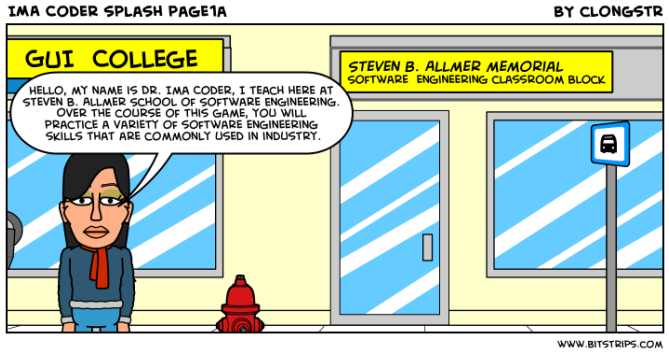
Animation movement 2
Effect: SPEAKING
Timing: MEDIUM</td><td></td></tr>
<tr><td colspan="3">Setting (visual)</td></tr>
<tr><td>Backdrop</td><td>College building, street view

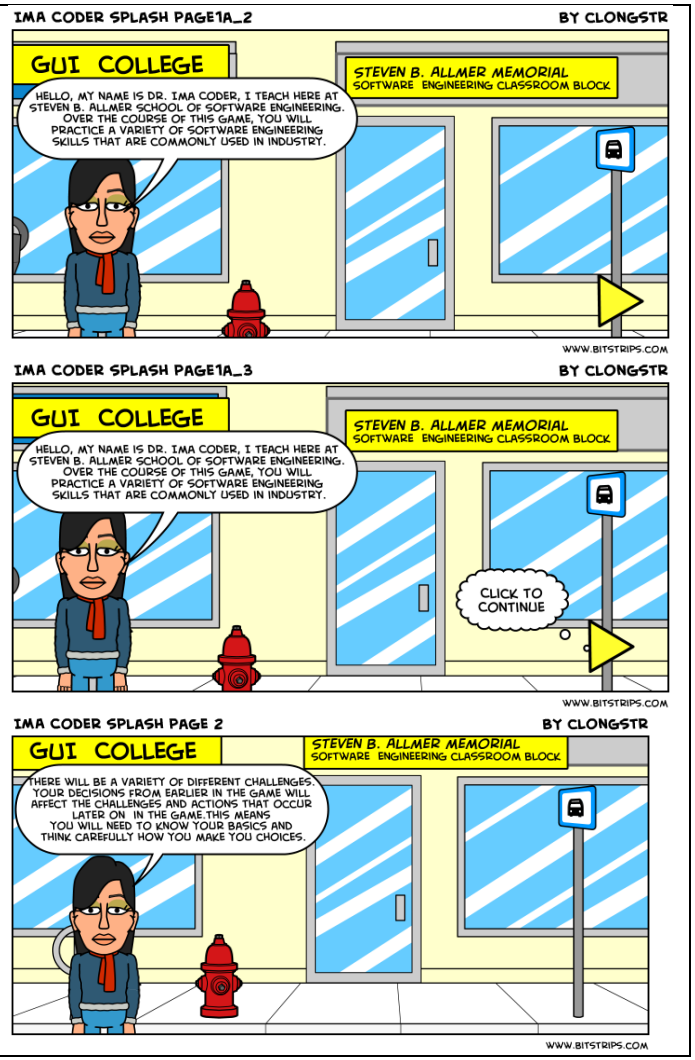
Note.
The building is a college; with GUI College sign and Steven B. Memorial Software Engineering Classroom block sign.</td><td></td></tr>
<tr><td colspan="3">Props</td></tr>
<tr><td>Generic interaction</td><td colspan="2"><table>
<tr><td>Conversation Bubble 1
Text: Hello, my name is Dr. Ima Coder, I teach here at Steven B. Allmer School of Software Engineering. Over the course of this game, you will practice a variety of software engineering skills that are commonly used in industry.
Speaker: Dr. Ima Coder
Size: LARGE
Location: USR</td><td>Right Arrow Button 1
Location: DSL

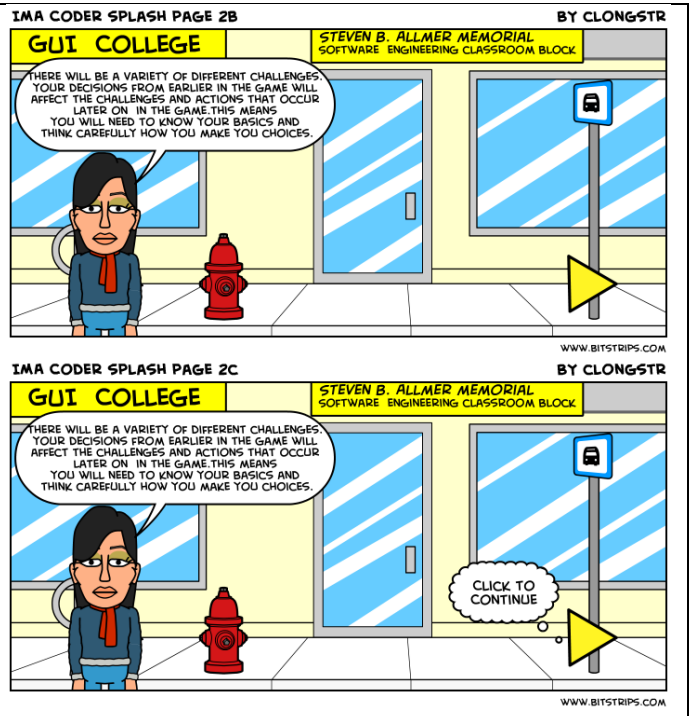
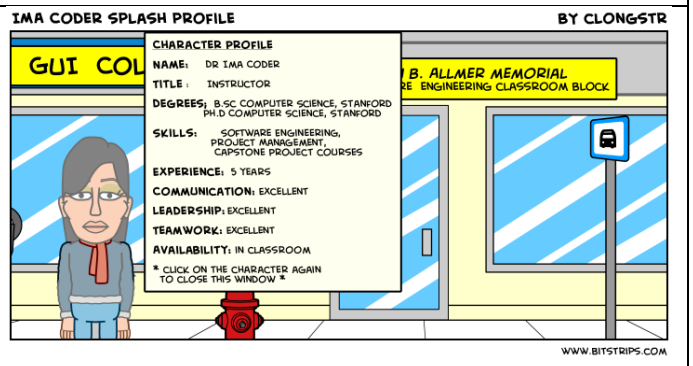
Right Arrow Button 2
Location: DSL</td><td>Hint Thought Bubble 1
Text: Click to continue
Speaker: Right Arrow Button 1
Size: SMALL
Location: CSL

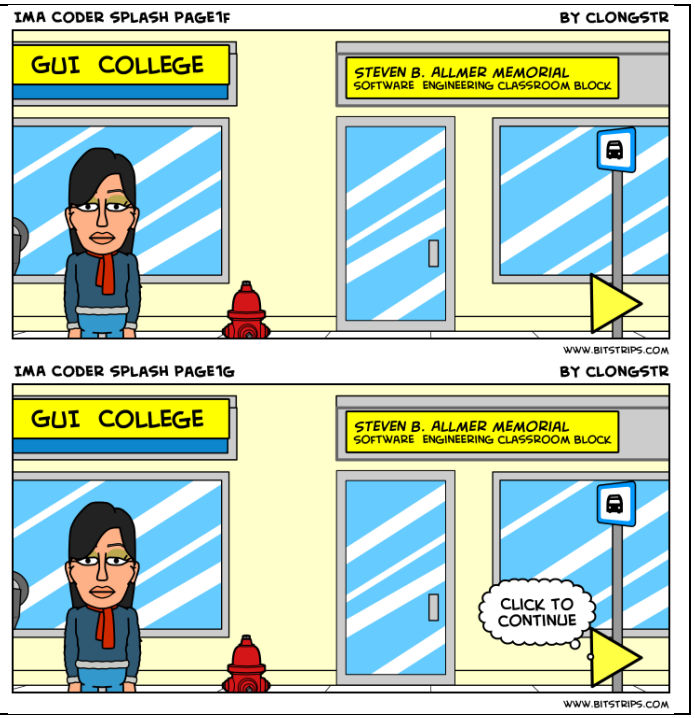
Hint Thought Bubble 2
Text: Click to continue
Speaker: Right Arrow Button 2
Size: SMALL
Location: CSL</td></tr>
</table></td></tr>
</table>

	Conversation Bubble 2 There will be a variety of different challenges that will be increasingly complex and your decisions from earlier in the game will affect the challenges and actions that occur later in the game. This means you will need to know your basics and think carefully about how you make your choices. Speaker: Dr. Ima Coder Size: LARGE Location: USR		
Set Decorations	Not present.		
Audio			
Music	Background music	 Opening Credits.mid	
Challenge	Not present		
Game Play			

Start of Screen	Backdrop College Building **Note. The College Building backdrop includes the bus stop sign, parking meter, fire hydrant.**	
Interactions (normal flow of events)	**Note. Start the background music for the screen.** START Background music **Note. The Dr. Ima Coder character enters and begins to speak.** START Dr. Ima Coder animation movement 1 FADE IN the Conversation Bubble 1 as a SLOW EFFECT Display for a MODERATE PRESENTATION amount of time. START Dr. Ima Coder animation effect 1 **Note. When Dr. Ima Coder has finished speaking, the right arrow is displayed, which allows the player to progress in the game by displaying the next conversation bubble.** FADE IN the Right Arrow Button 1 as a SLOW EFFECT If the player has not selected the arrow after HINT TIMER MODERATE amount of time, then display the Hint Thought Bubble 1.	

	Player clicks the Right Arrow Button 1 to progress in the game FADE OUT the Conversation Bubble 1 as a SLOW EFFECT FADE IN the Conversation Bubble 2 as a SLOW EFFECT Display for a MODERATE PRESENTATION amount of time. STOP Dr. Ima Coder animation effect 1 FADE OUT the Conversation Bubble 2 as a SLOW EFFECT **Note. When Dr. Ima Coder has finished speaking, the right arrow is displayed, which allows the player to progress in the game.** FADE IN the Right Arrow Button 2 as a SLOW EFFECT If the player has not selected the arrow after HINT TIMER MODERATE amount of time, then display the Hint Thought Bubble 2. Player clicks the Right Arrow Button 2 to progress in the game **Note. At the end of the screen, the game needs to stop the audio and remove the visual setting from the display.**	

		
Alternate flow of events	If an error occurs, then end the game. If the player selects the Dr. Ima Coder character, the character profile is displayed. If the player deselects the Dr. Ima Coder character, the character profile is removed from the display.	

End of Screen	Two options: Hint thought bubble not displayed Hint thought bubble displayed	

3.2.1.3 Act 1, Scene 1, Screen 3: Player Chooses Character

Table 14. Player Chooses Character

Identifier	Screen 3	
Purpose	The player is presented with a collection of avatar images. The player selects their avatar and names their character.	
Learning Objectives	Not present.	
Declarations, initialization		

<table>
<tr><td>Challenge</td><td>Not present.</td><td></td></tr>
<tr><td colspan="3">Characters</td></tr>
<tr><td>Player</td><td>Name: TBD
Pose: STANDING
Size: MEDIUM
Location: CS</td><td></td></tr>
<tr><td>Non-player characters</td><td>Not present.</td><td></td></tr>
<tr><td colspan="3">Setting (visual)</td></tr>
<tr><td>Backdrop</td><td>College student’s living room.

Note.
The college student living room backdrop includes the sofa, soccer ball, table, drink, clock, and picture.</td><td></td></tr>
<tr><td colspan="3">Props</td></tr>
<tr><td>Generic interaction</td><td colspan="2"><table>
<tr>
<td>Information Box 1
Text: Before you begin, you need to select at character. Choose your character from the following available avatars. Click on the arrow buttons to scroll through the different characters and then click on the character when you are ready to select it.
Size: MEDIUM
Location: USC

Information Box 2
Text: Choose your character
Size: SMALL
Location: CS</td>
<td>Note. For progressing in the screen
Right Arrow Button 1
Location: DSL

Note. For selecting character
Left Arrow Button 1
Location: CSL

Right Arrow Button 2
Location: CSR

Note. For confirming character selection
Left Arrow Button 2
Location: CSL</td>
<td>Thought Bubble 1
Text: Return to selection menu
Speaker: Left Arrow Button 2
Size: SMALL
Location: CSL

Hint Thought Bubble 2
Text: Confirm your choice and continue to give a name to your character
Speaker: Right Arrow Button 3
Size: SMALL
Location: CSL

Hint Thought Bubble 3
Text: Click the arrow to continue</td>
</tr>
</table></td></tr>
</table>

<table>
<tr><td></td><td>
<table>
<tr>
<td>Information Box 3
Text: You have selected this avatar for your character.

Text: Confirm by clicking the arrow to the right. You may return clicking the arrow on the left.
Size: MEDIUM
Location: DSR

Information Box 4
Text: Give your character a name. Click the arrow when you are finished.

Text: _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
Size: MEDIUM
Location: DSR

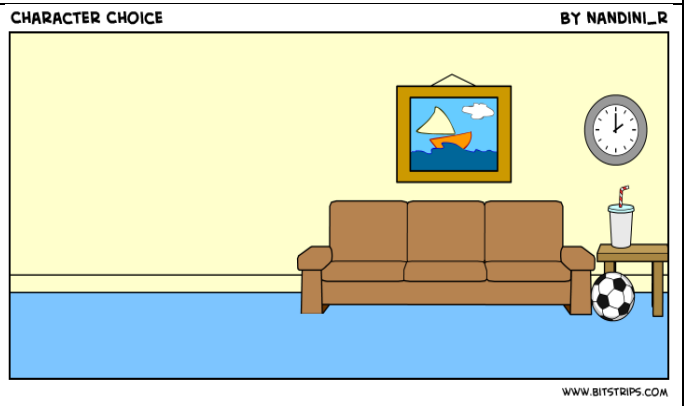
Information Box 5
If you are happy with your name click on the character to see your profile, or click the arrow to the next scene
Size: MEDIUM
Location: DSR</td>
<td>Right Arrow Button 3
Location: CSR

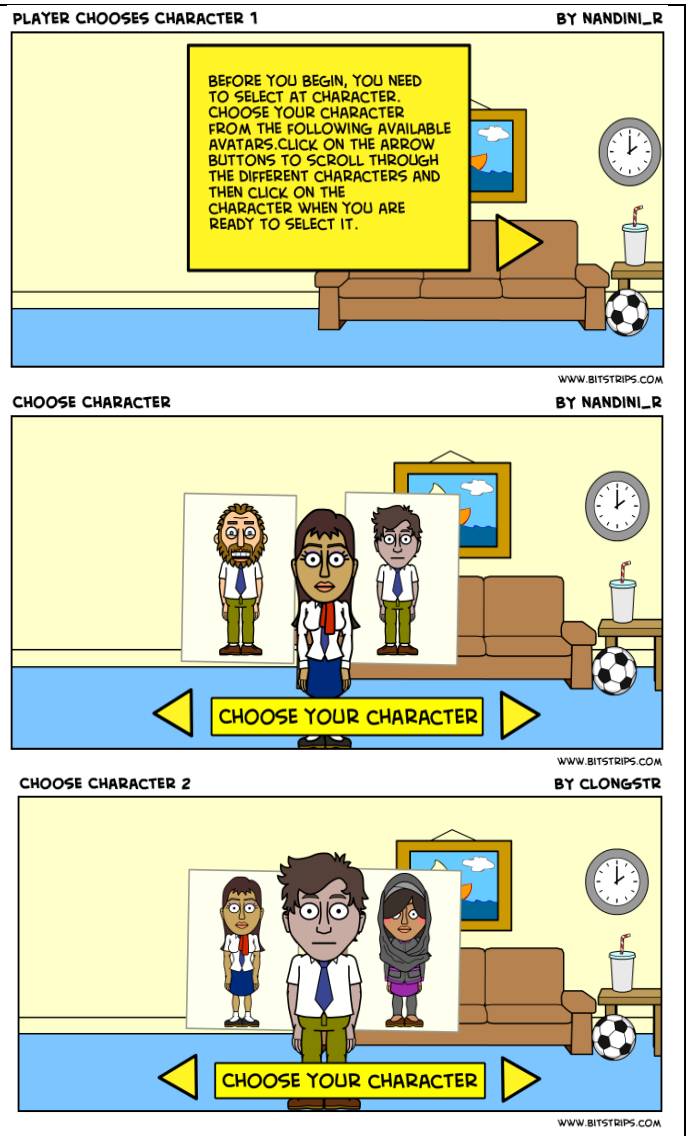
Note. For progressing to next scene
Right Arrow Button 4
Location: DSL</td>
<td>Speaker: Right Arrow Button 4
Size: SMALL
Location: CSL</td>
</tr>
</table>
</td></tr>
<tr><td>Set Decorations</td><td>
<table>
<tr>
<td>Name: Player Avatar 1
Size: MEDIUM
Location: CSR

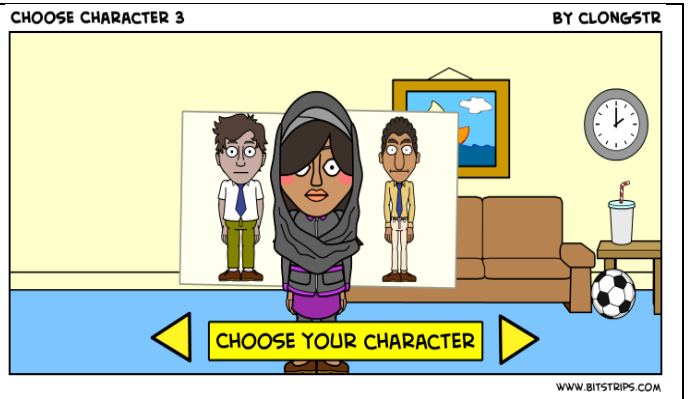
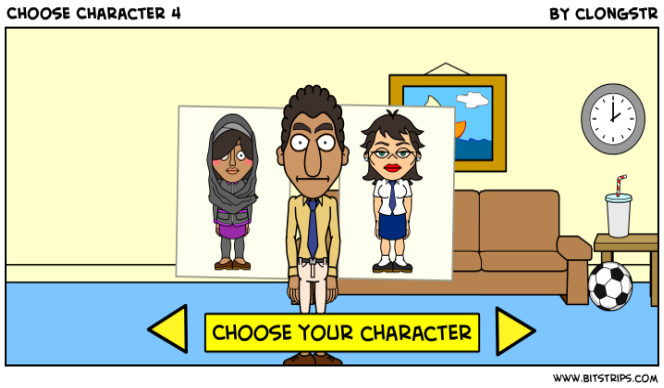
Name: Player Avatar 2
Size: MEDIUM
Location: CS</td>
<td>Name: Player 3
Size: MEDIUM
Location: CSL

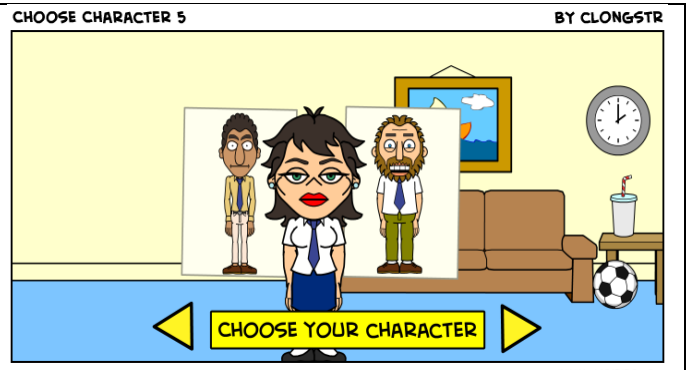
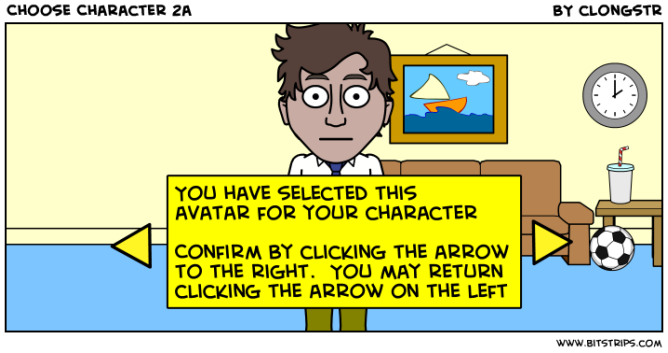
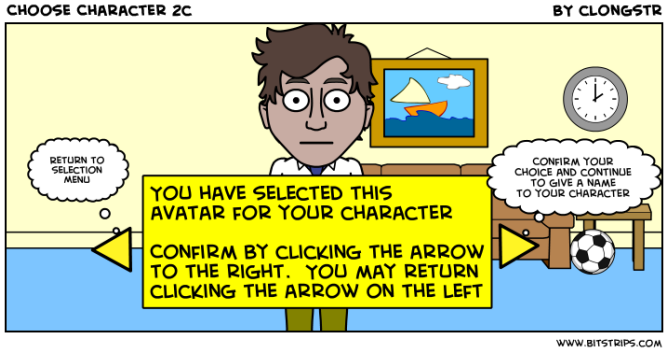
Name: Player Avatar 4
Size: MEDIUM
Location: OSR</td>
<td>Name: Player Avatar 5
Size: MEDIUM
Location: OSR

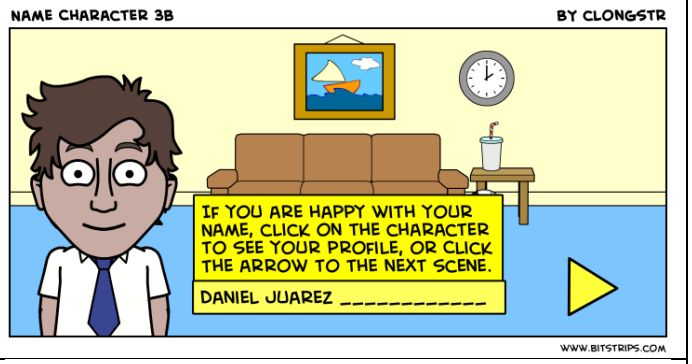
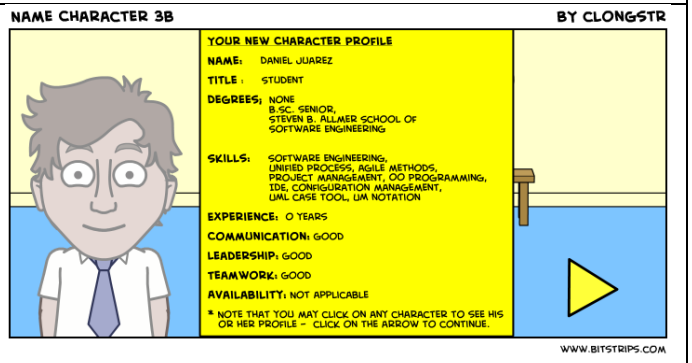
Name: Player Avatar 6
Size: MEDIUM
Location: OSR</td>
</tr>
</table>
</td></tr>
</table>

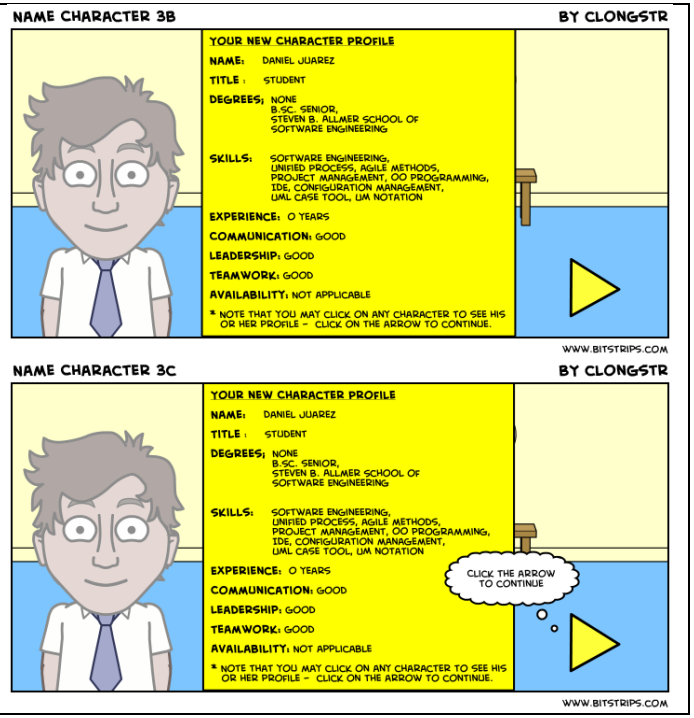
	Note. **The Player Avatars are presented to the player; player can select one and name their character in the game.**	
Audio		
Music	Background music	 PianoRoll1.mid
Challenge	Not present	
Game Play		
Start of Screen	1) Backdrop College Student Living Room	

Interactions (normal flow of events)	**Note. Start the background music for the screen.** START Background music **Note. Present the game play instructions in an information box.** FADE IN the Information Box 1 as a MEDIUM EFFECT Display the screen for a MODERATE PRESENTATION amount of time. **Note. When the last information box has been displayed for its presentation time then the right arrow is displayed, which allows the player to progress in the game.** FADE IN the Right Arrow Button 1 as a SLOW EFFECT Player clicks the Right Arrow Button 1 to proceed. FADE OUT the Information Box 1, Right Arrow Button 1 as a MODERATE EFFECT **Note. ** The player can select an avatar.** **All possible player avatars (Player 1..Player 6) can be displayed in the scene, as the player rotates through the options; the first three are displayed in a row.** FADE IN the Player 1, Player 2, and Player 3 as a FAST EFFECT FADE IN the Information Box 2, Left Arrow Button 1, and Right Arrow Button 2 as a FAST EFFECT Display the screen for a MODERATE PRESENTATION amount of time. If the player clicks the Left Arrow Button 1, then the characters displayed are shifted one to the left (e.g., if characters 4,5,6 are displayed and the Left Arrow Button 1 is clicked, then characters 5,6,1 are displayed.	

	If the player clicks the Right Arrow Button 2, then the characters displayed are shifted one to the right (e.g., if characters 1,2,3 are displayed and the Right Arrow Button 2 is clicked, then characters 6,1,2 are displayed. **Note. Player selects an avatar option and is prompted to confirm their selection.** **Note. This game is illustrated using Player 3 as the character player.** If the player selects an avatar, then FADE OUT the three avatars that are being displayed FADE OUT Left Arrow Button 1 FADE OUT Right Arrow Button 2 FADE IN Player as a QUICK EFFECT Size: MEDIUM Location: DSR FADE IN Information Box 3 as a SLOW EFFECT FADE IN the Left Arrow Button 2 as a SLOW EFFECT FADE IN the Right Arrow Button 3 as a SLOW EFFECT **Note. If the player selects the Left Arrow Button 1, then return to avatar selection.** FADE OUT Information Box 3 as a SLOW EFFECT FADE OUT the Left Arrow Button 2 as a SLOW EFFECT FADE OUT the Right Arrow Button 3 as a SLOW EFFECT Re-display the avatar selection part of the screen **. **Note. If player selects Right Arrow Button 2, then proceed to name the character.** FADE OUT Information Box 3 as a SLOW EFFECT FADE OUT the Left Arrow Button 2 as a SLOW EFFECT	 

	FADE OUT the Right Arrow Button 3 as a SLOW EFFECT FADE IN Information Box 4 as a FAST EFFECT FADE IN the Right Arrow Button 4 If player has not selected the right arrow after HINT TIMER MODERATE amount of time, then display Hint Thought Bubble 3. Player clicks the Right Arrow Button 4 to end the screen. **Note. At the end of the screen, the game needs to stop the audio and remove the visual setting from the display.**	  

		 
Alternate flow of events	If an error occurs, then end the game. If the player selects a character, the character profile is displayed. If the player deselects a character, the character profile is removed from the display. Note. The character may be the player or non-player character.	
End of Screen	Two options:	

	Hint thought bubble not displayed Hint thought bubble displayed	

3.2.1.4 Act 1, Scene 1, Screen 4: Nim Esis

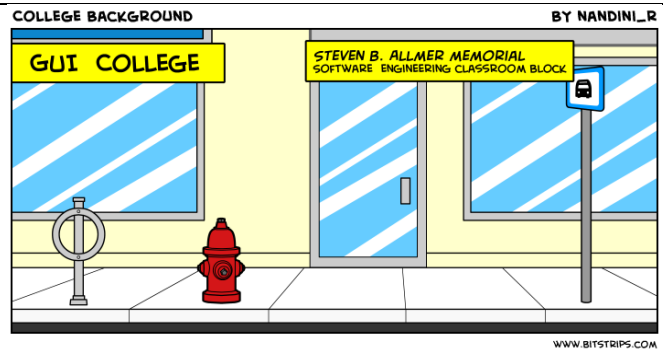
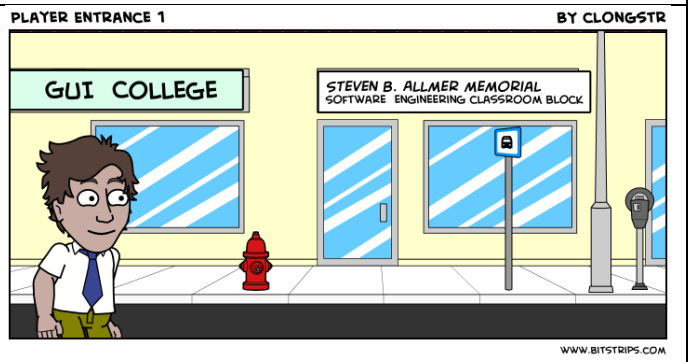
Table 15. Introduce Nim Esis

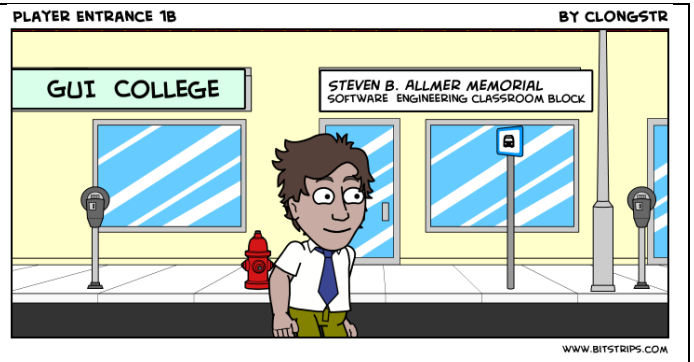
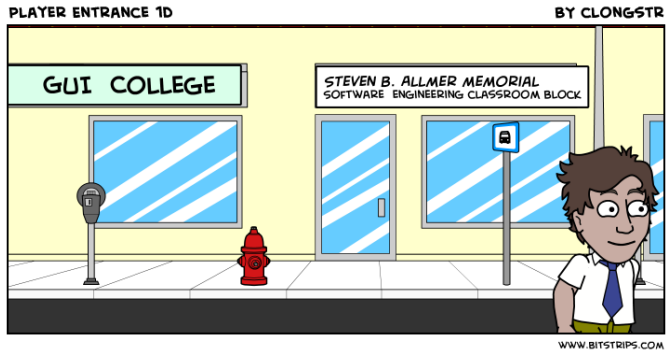
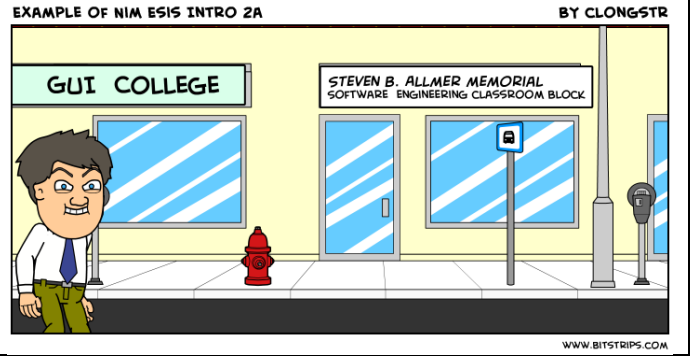
Identifier	Screen 4 Note. Introduce Nim Esis character.	
Purpose	Introduce Nim Esis, one of the main non-player characters in the game. Nim Esis is a student at the college. His character type from the ontology is antagonist.	
Learning Objectives	Not present.	

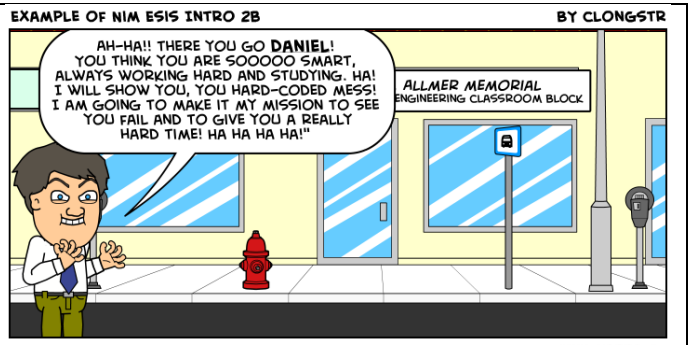
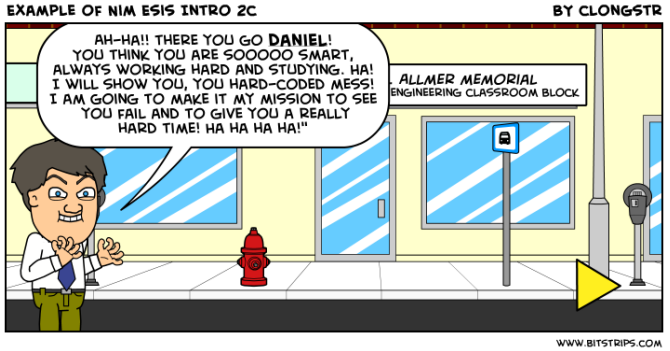
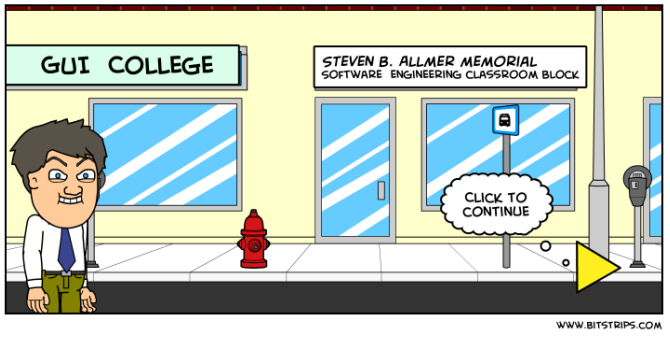
Declarations, initialization		
Challenge	Not present.	
Characters		
Player	Player: <named in previous screen by the player> Pose: STANDING Size: MEDIUM Location: OSR Animation movement 1 Movement: GLIDE Timing: SLOW Loop: NO Move to: DSL Path: STRAIGHT	
Non-player characters	NPC: Nim Esis Pose: STANDING Size: MEDIUM Location: OSR Animation movement 1 Effect: SPEAKING Timing: MEDIUM Animation movement 2 Effect: WALKING Timing: MEDIUM Move to: OSL Path: STRAIGHT	
Setting (visual)		
Backdrop	College building, street view	

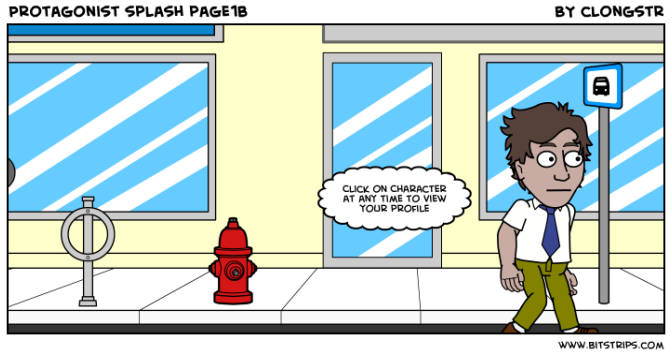
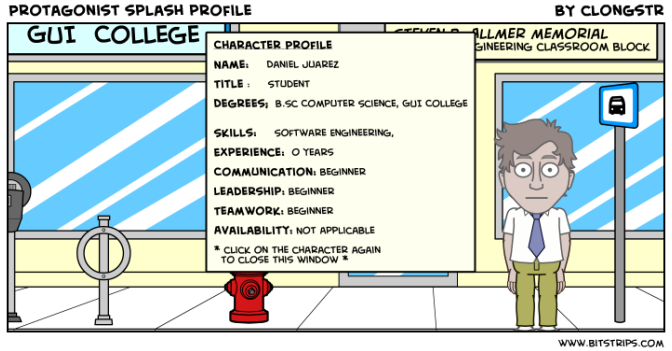
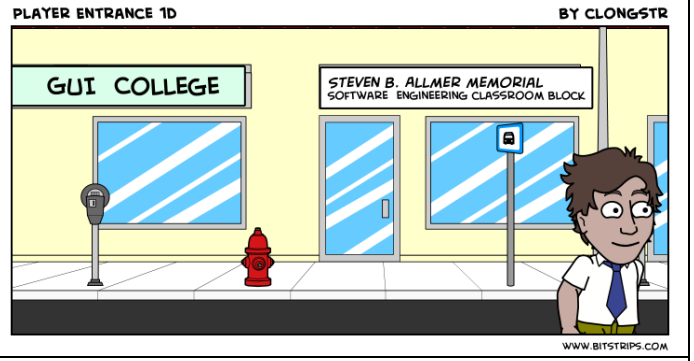
<table>
<tr><td></td><td colspan="2">Note.
The building is a college; with GUI College sign and Steven B. Memorial Software Engineering Classroom block sign.</td><td></td></tr>
<tr><td colspan="4">Props</td></tr>
<tr><td>Generic interaction</td><td colspan="3"><table><tr><td>Name: Conversation Bubble 1
Text: Ah, there you are <player name>. You think you are soooo smart, always working hard and studying. Ha! I will show you, you hard-coded mess! I am going to make it my mission to see you fail and give you a really hard time! Ha! Ha! Ha! Ha!
Size: MEDIUM
Speaker: Nim Esis
Location: DSR</td><td>Name: Right Arrow Button 1
Location: DSL</td><td>Name: Hint Thought Bubble 1
Text: Click to continue
Speaker: Right Arrow Button 1
Size: SMALL
Location: CSL

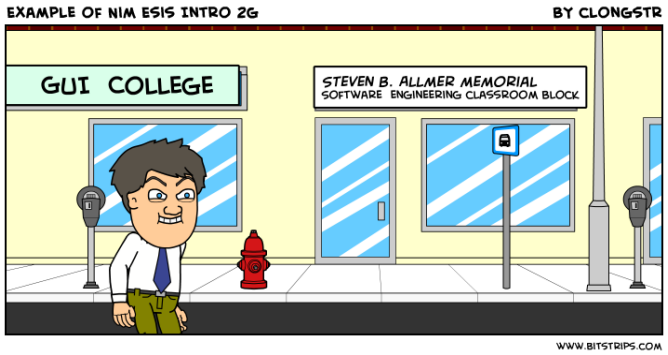
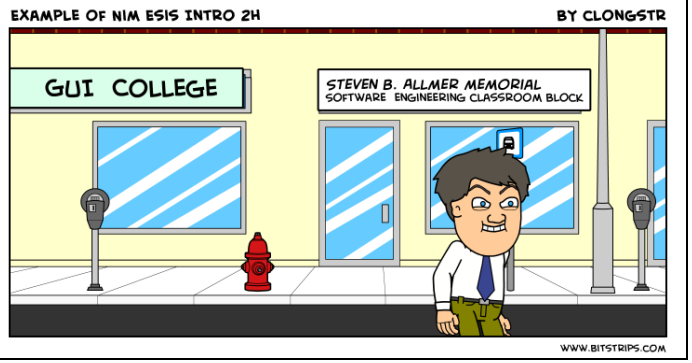
Click arrow to move to the next scene?</td></tr></table></td></tr>
<tr><td>Set Decorations</td><td colspan="3">Not present.</td></tr>
<tr><td colspan="4">Audio</td></tr>
<tr><td>Music</td><td colspan="2">Background music</td><td> 
PianoRoll1.mid</td></tr>
<tr><td>Challenge</td><td colspan="2">Not present</td><td></td></tr>
<tr><td colspan="4">Game Play</td></tr>
</table>

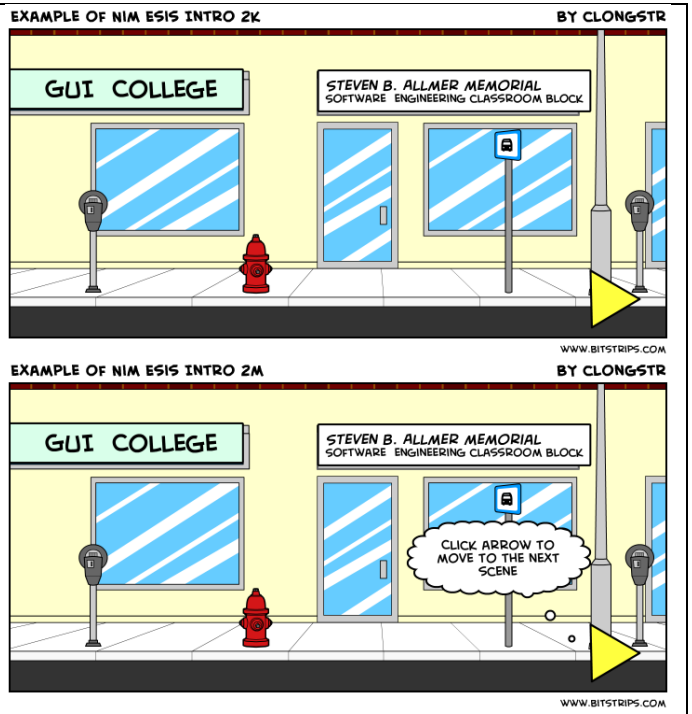
Start of Screen	1) Backdrop College Building **Note. The College Building backdrop includes the bus stop sign, parking meter, fire hydrant.**	
Interactions (normal flow of events)	**Note. Start the background music for the screen.** START Background music **Note. The player character enters, then moves across the display.** FADE IN Player as a QUICK EFFECT Size: MEDIUM Location: DSR START Player animation effect 1 **Note. The Nim Esis character enters, then introduces himself.** FADE IN Nim Esis as a QUICK EFFECT Size: MEDIUM Location: DSR FADE IN the Conversation Bubble 1 as a QUICK EFFECT START Nim Esis animation movement 1 Display the screen for a MODERATE PRESENTATION amount of time. STOP Nim Esis animation movement 1 FADE OUT the Conversation Bubble 1 as a SLOW EFFECT	

	Note. As Nim Esis walks off, the right arrow is displayed, which allows the player to progress in the game. START Nim Esis animation movement 2 FADE IN the Right Arrow Button 1 as a SLOW EFFECT If player has not selected the Right Arrow Button 1 after HINT TIMER MODERATE amount of time, then display Hint Thought Bubble 1. Player clicks the Right Arrow Button 1 to end the screen. **Note. At the end of the screen, the game needs to stop the audio and remove the visual setting from the display.**	  

		  
Alternate flow		

of events	If an error occurs, then end the game. If the player selects a character, the character profile is displayed. If the player deselects a character, the character profile is removed from the display. **Note. The character may be the player or non-player character.**	  
End of Screen	Two options:	

	Hint thought bubble not displayed Hint thought bubble displayed	 

3.2.2 Act 1, Scene 2 – Classroom Challenge

Table 16. Scene 2 – Classroom Challenge

Identifier	Scene 2
Purpose	The player and Nim Esis are in a classroom answering questions from the instructor, Ima Coder.
Learning objectives	Challenges the student on her/his general, background knowledge of: Software Testing Challenge = Interactive Quiz SWEBOK = Software Testing Software Testing Fundamentals Test Techniques

	Bloom = Knowledge, Comprehension Software Design Challenge = Interactive Quiz SWEBOK = Software Design General Design Concepts Context of Software Design Software Design Process Bloom = Knowledge, Comprehension Software Engineering Process Challenge = Interactive Quiz SWEBOK = Software Engineering Process Process Definition Bloom = Knowledge, Comprehension If the player is successful in answering, she/he will be able to move forward in the game and collect points towards a future reward.
Uses Screens	Screen1 Classroom Quiz Introduction, quiz, wrap-up
Behaviour state machine description: Current Transition event condition output Next	Initial state for Scene 1 FADE IN Screen 1 in a MODERATE AMOUNT OF TIME WAIT MODERATE AMOUNT OF TIME FADE OUT Screen 1 in a MODERATE AMOUNT OF TIME Final state for Scene 1
Alternate flow of events	If an error occurs, then end the game.

3.2.2.1 Act 1, Scene 2, Screen 1 – Classroom Quiz Challenge (introduction, quiz, wrap-up)

Table 17. Classroom Quiz Challenge

Identifier	Screen 1	

Purpose	The player and Nim Esis are in a classroom answering questions from the professor, Ima Coder. Ima Coder starts the lesson by informing the students about an upcoming job fair, and that if they answer the majority of her questions correctly she will offer her recommendation. Likewise, if all the questions are answered correctly they will get the school's highest recommendation.	
Learning Objectives	Challenges the student on her/his general, background knowledge of: Software Testing Challenge = Interactive Quiz SWEBOK = Software Testing Software Testing Fundamentals Test Techniques Bloom = Knowledge, Comprehension Software Design Challenge = Interactive Quiz SWEBOK = Software Design General Design Concepts Context of Software Design Software Design Process Bloom = Knowledge, Comprehension Software Engineering Process Challenge = Interactive Quiz SWEBOK = Software Engineering Process Process Definition Bloom = Knowledge, Comprehension If the player is successful in answering, she/he will be able to move forward in the game and collect points towards a future reward.	
Declarations, initialization		

Challenge	Not present.	
Characters		
Player	Player: <named in previous screen by the player> Pose: SITTING AT DESK Size: MEDIUM Location: CSL	
Non-player characters	NPC: Dr. Ima Coder Pose: STANDING Size: MEDIUM Location: OSR Animation movement 1 Effect: SPEAKING Timing: MEDIUM Character1StandClosed.png Character1StandOpen.png Animation movement 2 Effect: WALKING Timing: MEDIUM Character1WalkRLeftClosed.png Character1WalkRRightClosed.png NPC: Nim Esis Pose: SITTING AT DESK Size: MEDIUM Location: CSR	
Setting (visual)		
Backdrop	1) Backdrop College Classroom **Note. College classroom backdrop includes wall**	

<table>
<tr><td></td><td colspan="2">clock, teacher’s desk, blackboard.</td><td></td></tr>
<tr><td>Props</td><td colspan="3"></td></tr>
<tr><td>Generic interaction</td><td colspan="3"><table>
<tr>
<td>Conversation Bubble 1
Text: Good morning <player name> ! This morning, I want you and Nim Esis to know about this new opportunity that has come up. There is a recruiting fair coming up next week. You are both very promising students and I want to make sure that you are ready for this opportunity.
Size: MEDIUM
Speaker: Dr. Ima Coder
Location: CSR

Conversation Bubble 2
Text: Nim has already passed the last exam series in the last class. But there is one more spot open for a promising student to attend the recruiting fair.
Size: MEDIUM
Speaker: Dr. Ima Coder
Location: CSR

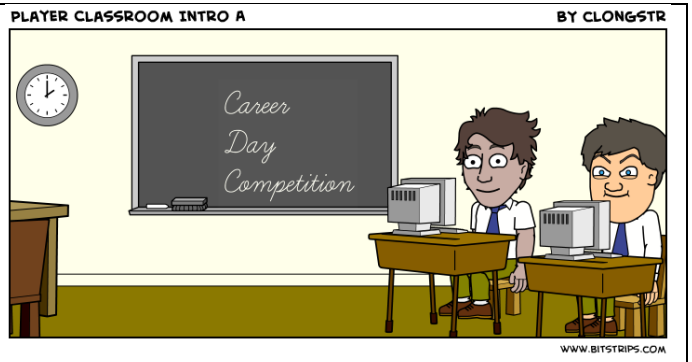
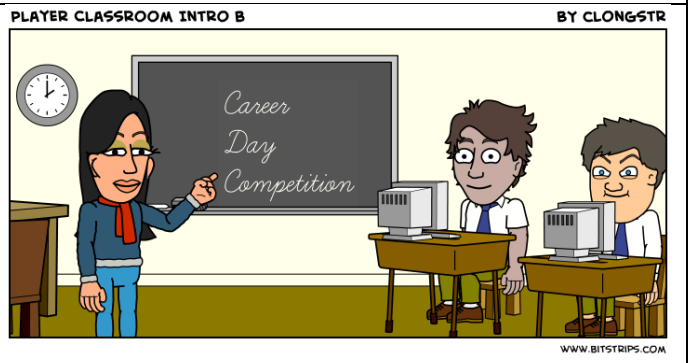
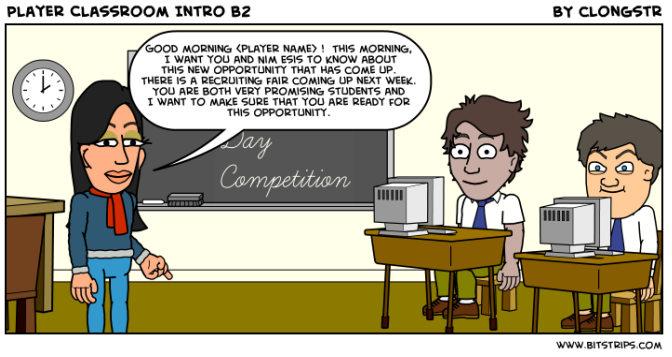
Conversation Bubble 3
Text: <Player Name>, in order for you to move on to the recruiting fair, I am going to ask you a series of questions that every software engineer manager should know.
Size: MEDIUM
Speaker: Dr. Ima Coder</td>
<td>Button 1
Text: Click here to continue
Location: DSC

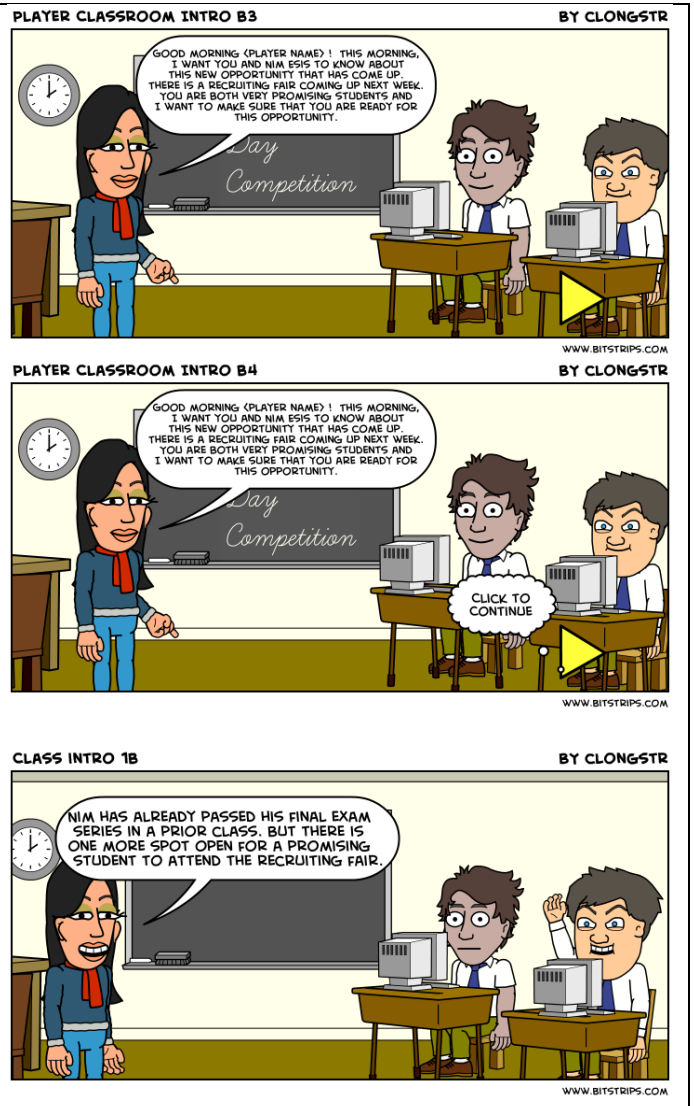
Button 2
Text: Click here to continue
Location: DSC

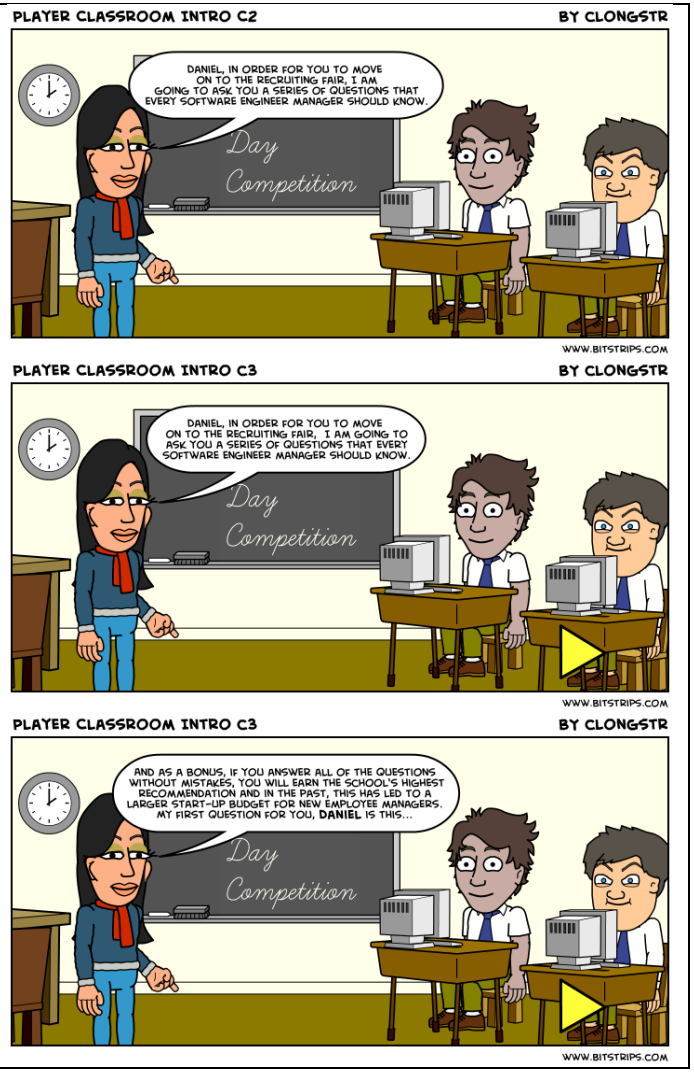
Button 3
Text: Click here to continue
Location: DSL

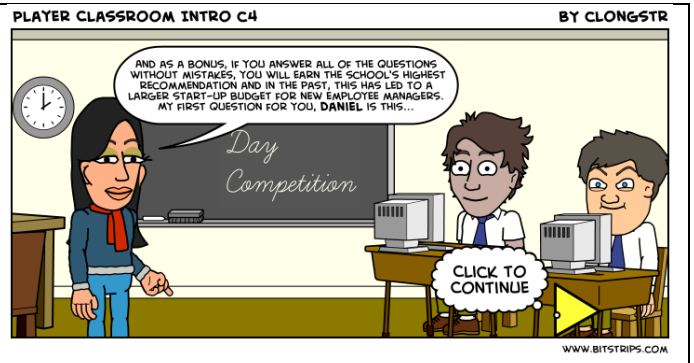
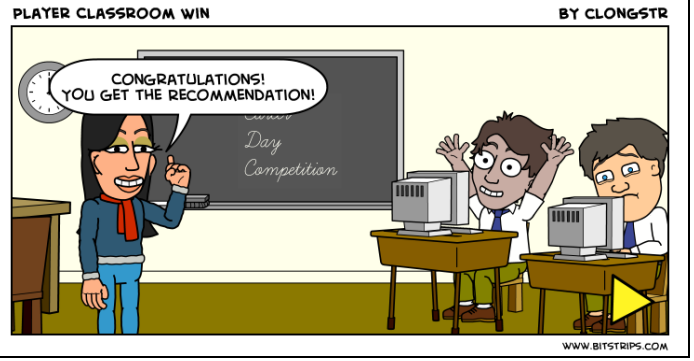
Right Arrow Button 1
Location: DSL</td>
<td>Hint Thought Bubble 1
Text: Select the arrow button to proceed
Speaker: Right Arrow Button 1
Size: SMALL
Location: CSL</td>
</tr>
</table></td></tr>
</table>

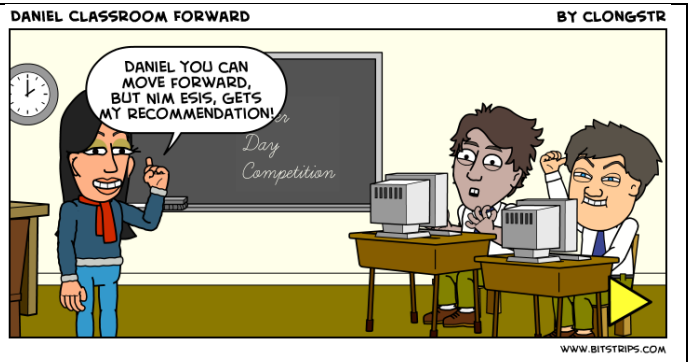
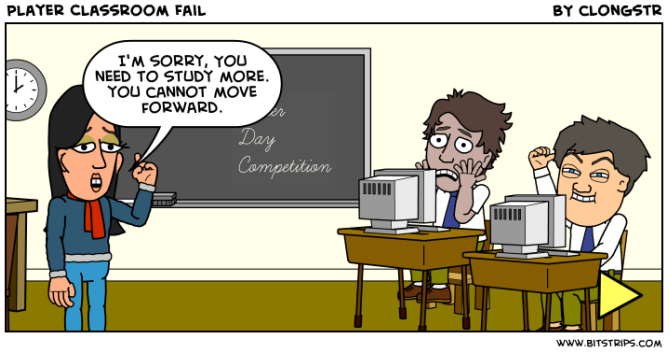
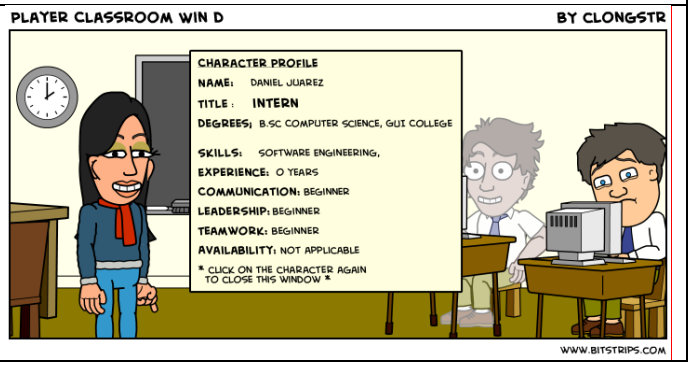
	Location: CSR Conversation Bubble 4 Text: And as a bonus, if you answer all of the questions without mistakes, you will earn the school's highest recommendation and in the past, this has led to a larger start-up budget for new employee managers. My first question for you, <Player Name> is this... Size: MEDIUM Speaker: Dr. Ima Coder Location: CSR		
Set Decorations	Not present.		
Audio			
Music	Background music	 PianoRoll1.mid	
Challenge	Quiz 1		
Game Play			
Start of Screen	1) Backdrop College Classroom + Player and Nim Esis at their desks		

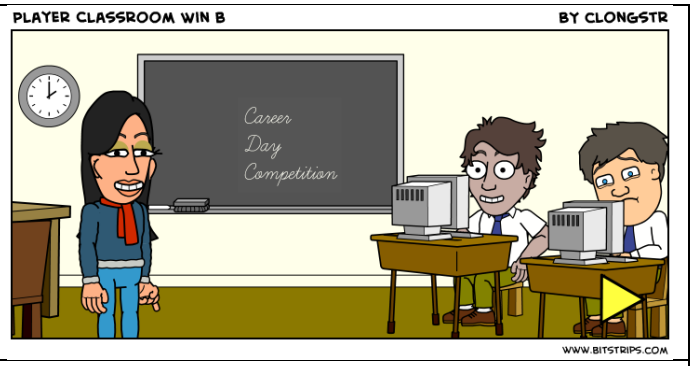
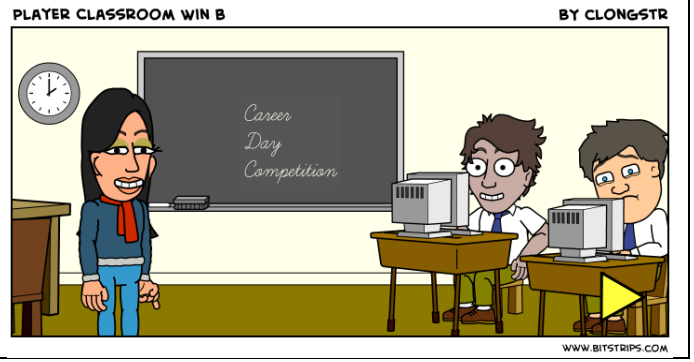
		
Interactions (normal flow of events)	**Note. Start the background music for the screen.** START Background music **Note. Dr. Ima Coder walks into the classroom and explains the quiz challenge.** START Dr. Ima Coder animation effect 1 FADE IN the Conversation Bubble 1 as a QUICK EFFECT FADE IN the Button 1 as a QUICK EFFECT START Dr. Ima Coder animation effect 2 Display the screen for a MODERATE PRESENTATION amount of time. STOP Dr. Ima Coder animation effect 2 FADE OUT the Conversation Bubble 1 as a SLOW EFFECT FADE IN the Conversation Bubble 2 as a QUICK EFFECT FADE IN the Button 2 as a QUICK EFFECT START Dr. Ima Coder animation effect 2 Display the screen for a MODERATE PRESENTATION amount of time. STOP Dr. Ima Coder animation effect 2	 

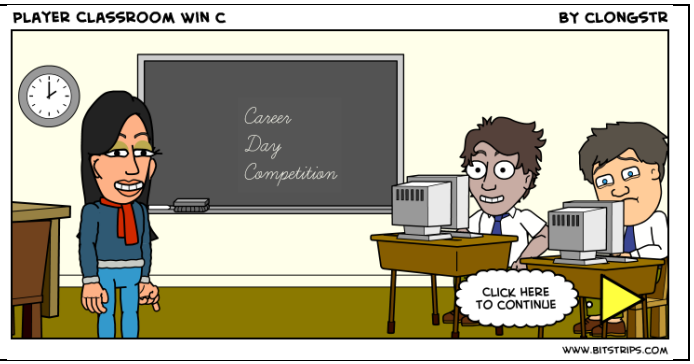
	FADE OUT the Conversation Bubble 2 as a SLOW EFFECT FADE IN the Conversation Bubble 3 as a QUICK EFFECT FADE IN the Button 3 as a QUICK EFFECT START Dr. Ima Coder animation effect 2 Display the screen for a MODERATE PRESENTATION amount of time. STOP Dr. Ima Coder animation effect 2 FADE OUT the Conversation Bubble 3 as a SLOW EFFECT FADE IN the Conversation Bubble 4 as a QUICK EFFECT START Dr. Ima Coder animation effect 2 Display the screen for a MODERATE PRESENTATION amount of time. STOP Dr. Ima Coder animation effect 2 FADE OUT the Conversation Bubble 4 as a SLOW EFFECT **Note. When Dr. Ima Coder has finished speaking, the continue button is displayed, which allows the player to progress in the game.** FADE IN the Button 1 as a SLOW EFFECT Player clicks the Button 1 to progress in the screen. Note. Play Quiz 1 START Quiz 1 **Note. Quiz 1 wrap-up** At the end of the set of questions the student is assessed using the following rule set: If the player accumulates > 200 points, then the student wins the school recommendation If the player accumulates 100..200 points, then the student will be allowed to progress but Nim Esis gets the school	

	recommendation. If the player accumulates <100 points, then the student will have to restart the game. **Note. When the quiz has been played, then the right arrow is displayed, which allows the player to progress in the game.** FADE IN the Right Arrow Button 1 as a SLOW EFFECT If player has not selected the Right Arrow Button 1 after HINT TIMER MODERATE amount of time, then display Hint Thought Bubble 1. Player clicks the Right Arrow Button 1 to end the screen. **Note. At the end of the screen, the game needs to stop the audio and remove the visual setting from the display.**	

		 **Note. See Quiz 1 for Storyboard figures.** 

		 
Alternate flow of events	If an error occurs, then end the game. If the player selects a character, the character profile is displayed. If the player deselects a character, the character profile is removed from the display. **Note. The character may be the player or non-player character.**	

		
End of Screen	Two options: Hint thought bubble not displayed Hint thought bubble displayed	

3.2.2.1.1 Act 1, Scene 2, Screen 1, Quiz 1

Table 18. Quiz 1

Identifier	Quiz 1	
Purpose	This quiz challenges the player on fundamental SE concepts including: requirements; architecture; testing; and process. The level of difficulty of the questions is average.	
Learning Objectives	Challenges the student on her/his general, background knowledge of: Software Testing Challenge = Interactive Quiz SWEBOK = Software Testing Software Testing Fundamentals Test Techniques Bloom = Knowledge, Comprehension Software Design Challenge = Interactive Quiz SWEBOK = Software Design General Design Concepts Context of Software Design Software Design Process Bloom = Knowledge, Comprehension	

	Software Engineering Process Challenge = Interactive Quiz SWEBOK = Software Engineering Process Process Definition Bloom = Knowledge, Comprehension If the player is successful in answering, she/he will be able to move forward in the game and collect points towards a future reward.	
Quiz elements	**Note. The quiz has 4 main questions.** **Note. Question 1** Quiz question Q1 Concept: Design SWEBOK: General Design Concepts SWEBOK: Context of Software Design Level of Difficulty: Average Text: What does a design specification describe? Hint: None Quiz answer Q1.A1 A specification of what will be implemented Evaluation: INCORRECT Feedback: No, I'm afraid not, <player name>. Recall that a requirements specification is used to define what will be implemented. Let's try another question. Quiz answer Q1.A2 Text: A specification of how the software will be implemented. Evaluation: CORRECT Feedback: Yes, you're correct! You are on your way to earning a higher recommendation from me and from the college. As a bonus, <Player name>, you have earned some start-up funds after you graduate from this course!	

	Quiz answer Q1.A3 Text: A specification of the GUI Evaluation: OK Feedback: Hmmm, alright, <Player name>, your answer has some merit. **Note. Response requires a follow-up question** **Note. Question 1 – follow-up question** Quiz Question Q1F Concept: Design Level of Difficulty: Average Text: Why did you choose this answer? Quiz answer Q1F.A1 Text: Is it because the design specification can include the GUI specification? Evaluation: CORRECT Feedback: Yes! That's it. Alright, let's move on to another question. Quiz answer Q1F.A2 Text: Is it because the GUI specification is always the most important part of a design specification? Evaluation: INCORRECT Feedback: No, that is not the case here. I'm sorry. Maybe you will have more success with the next question. Afterwards, though, perhaps you can go back and review this component of software engineering in your notes or textbook. Quiz answer Q1F.A3 Text: Is it because the GUI specification is always the most complex part of a design specification? Evaluation: INCORRECT Feedback: No, that is not the case here. I'm sorry. Maybe you will	

	have more success with the next question. Afterwards, though, perhaps you can go back and review this component of software engineering in your notes or textbook. **Note. Question 2** Quiz question Q2 Concept: Testing SWEBOK: Software Testing Fundamentals SWEBOK: Test Techniques Level of Difficulty: Average Text: Ok, now let me ask you another question: What does black box testing mean? Hint: None Quiz answer Q2.A1 Text: A test with access to internal data structures and algorithms Evaluation: INCORRECT Feedback: No, I'm afraid not. Recall that whitebox testing uses knowledge of internal data structures and algorithms. This is probably something you can review when you are studying this weekend. Quiz answer Q2.A2 Text: A test without any knowledge of internal implementation. Evaluation: CORRECT Feedback: Yes! That's it. Alright, let's move on to another question. Quiz answer Q2.A3 Text: A test that seeks to verify the interfaces between components Evaluation: OK Feedback: This is on the right track, but I need some clarification **Note. Response requires a follow-up question**	

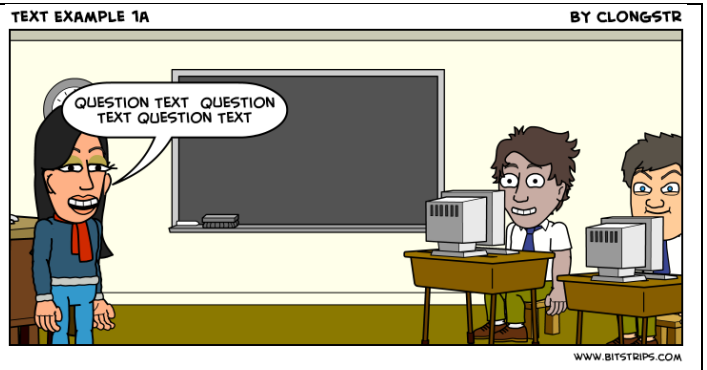
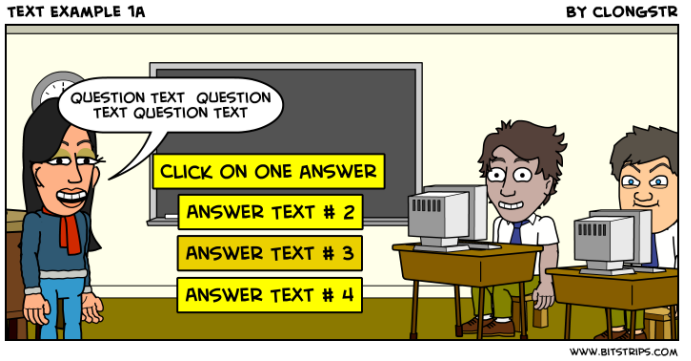
	Note. Follow-up question Quiz Question Q2F Concept: Testing Level of Difficulty: Average Text: Why did you choose this answer? Quiz answer Q2F.A1 Text: Is it because blackbox testing can be used on artifacts at different levels of abstraction (system, integration, unit); verifying interfaces is part of integration testing? Evaluation: CORRECT Feedback: Good! Quiz answer Q2F.A2 Text: Is it because blackbox testing is the most commonly used approach to verify component interactions? Evaluation: INCORRECT Feedback: No, that is not the case here. I'm sorry. Maybe you will have more success with the next question. Afterwards, though, perhaps you can go back and review this component of software engineering in your notes or textbook **Note. Question 3** Quiz question Q3 Concept: Architecture SWEBOK: General Design Concepts SWEBOK: Context of Software Design SWEBOK: Software Design Process Level of Difficulty: Average Text: Alright, let's press on. For large systems involving a lot of people and a lot of software components, what is the main purpose of a software architecture? Quiz answer Q3.A1	

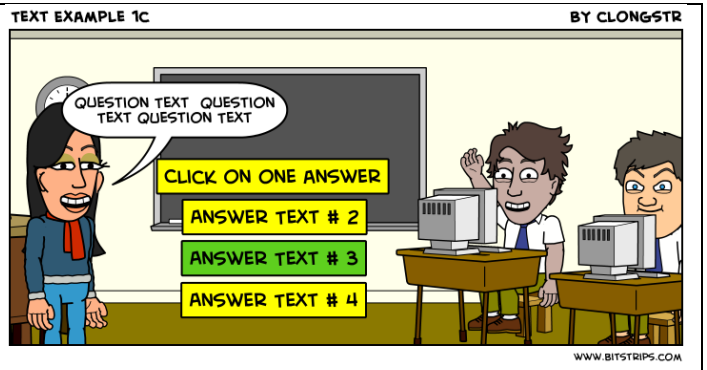
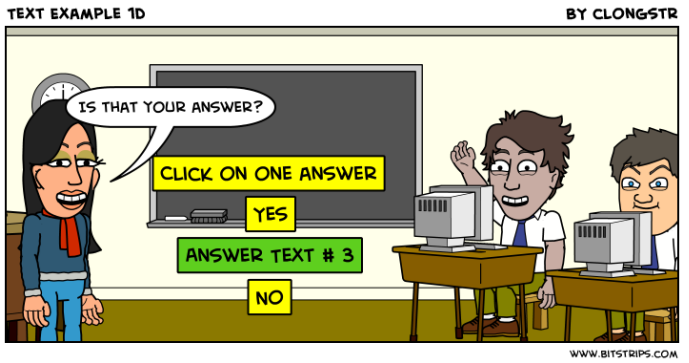
	Text: Specify the overall structure and behavior or the software system Evaluation: CORRECT Feedback: "Well done! That is correct; the main purpose of a software architect in that context is to specify the overall structure and behavior of the software system" Quiz answer Q3.A2 Text: Specify the domain model of the system. Evaluation: INCORRECT Feedback: I'm afraid not, remember that a domain model is typically done as part of the requirements engineering activity. Let's move on to the last question. Quiz answer Q3.A3 Text: Refine the requirements Evaluation: INCORRECT Feedback: "That is not correct, recall that this is not the main purpose of a software architecture, although. Let's move on to the last question" Quiz answer Q3.A4 Text: Specify the all the classes, signatures, data attributes to be implemented with visibility Evaluation: INCORRECT Feedback: Your answer is off the mark. Recall that this level of design is typically done in the detailed design activity, not the software architecture. Let's move on to the last question. **Note. Quiz Question 4** Quiz question Q4 Concept: Process SWEBOK: Process Definition Level of Difficulty: Average	

	Text: My last question is this, what are the three biggest strengths of scrum method of software development team management? Hint: None Quiz answer Q4.A1 Text: Scrum is the best approach for project with stable and well-defined requirements. Evaluation: INCORRECT Feedback: I'm sorry, that is not quite it. Recall that a plan-driven, waterfall approach is better suited for projects with stable and well-defined requirements. Quiz answer Q4.A2 Text: Scrum is a good approach for dealing with changes. Evaluation: CORRECT Feedback: Looks like you've been studying. That is the CORRECT answer, scrum method does help manage software development teams stay agile in the face of frequent changes. Now let's tally your score. Quiz answer Q4.A3 Text: Scrum has a low overhead cost in terms of process and management. Evaluation: OK Feedback: This is on the right track. **Note. Response requires a follow-up question** **Next question : Quiz Question Q4F1** Quiz answer Q4.A4 Text: Scrum has long development cycles Evaluation: INCORRECT Feedback: Recall that Scrum, an agile method, typically uses short iterations (sprints).	

	Quiz answer Q4.A5 Text: Scrum makes it easier to deliver a quality product in a scheduled time. Evaluation: OK Feedback: This is on the right track. **Note. Response requires a follow-up question** **Next question : Quiz Question Q4F2** **Note. Quiz Question 4 follow-up 1** Quiz Question Q4F1 Concept: Process Level of Difficulty: Average Text: Why did you choose this answer? Quiz answer Q4F1.A1 Text: Scrum always has a low overhead cost in terms of process and management" Evaluation: INCORRECT Feedback: Sorry, this is not correct for all projects. Quiz answer Q4F1.A2 Text: For projects with specific characteristics (e.g., smaller projects, co-located development team, well-defined requirements), Scrum can have a low overhead cost in terms of process and management. Evaluation: CORRECT Feedback: Good work! **Note. Quiz Question 4 follow-up 2** Quiz question Q4F2 Concept: Process Level of Difficulty: Average Text: Why did you choose this answer? Hint: None	

	Q4F2.A1 Text: For projects with specific characteristics (e.g., smaller projects, co-located development team, well-defined requirements) Scrum can make it easier to deliver a quality product in a scheduled time. Evaluation: CORRECT Feedback: Yes, that's right! Q4F2.A2 Text: Scrum always makes it easier to deliver a quality product in a scheduled time. Evaluation: INCORRECT Feedback: This is not correct for all projects.	
Reward scheme	Correctly answered question: add 100 points Correctly answered on follow-up question: add 50 points Incorrectly answered question or follow-up question: none	
Declarations, initialization		
Character presenting the quiz	Dr. Ima Coder	
Characters taking the quiz	Player Nim Esis	
Quiz style	**Note. Quiz style declarations** Question text displayed in a Conversation Bubble. Answer text displayed in buttons; player or NPC selects the button to answer the question. Feedback text displayed in a Conversation Bubble. Player selects a right arrow button to progress in the game.	

	Quiz Layout: Character asking questions, providing feedback is on the right side of the stage. Answers to quiz questions at the center of the stage, in a vertical alignment. Question Type: Conversation Bubble Size: MEDIUM Speaker: Dr. Ima Coder Colour: WHITE Instruction Type: Information Box Size: MEDIUM Colour: LIGHT YELLOW Answer Type: Button Size: MEDIUM Colour: LIGHT YELLOW Hint Type: Hint Thought bubble Text: Choose an answer before Nim Esis raises his hand! Speaker: Information box Size: SMALL Colour: WHITE Confirmation (yes, no) Type: Button Size: MEDIUM Colour: LIGHT YELLOW	 

	Feedback Type: Conversation Bubble Size: MEDIUM Speaker: Dr. Ima Coder Colour: WHITE Progression Type: Right Arrow Button Size: SMALL Colour: LIGHT YELLOW Hint Type: Hint Thought bubble Text: Click here to continue… Speaker: Right Arrow Button Size: SMALL Colour: WHITE **Note. Quiz behavior** **Note. Question** FADE IN Question prop as a QUICK EFFECT START character giving the quiz animation effect SPEAKING Display the screen for a MODERATE PRESENTATION amount of time. STOP character giving the quiz animation effect SPEAKING **Note. Answer** FADE IN Answer prop as a QUICK EFFECT Display the screen for a MODERATE PRESENTATION amount of time. If player has not selected an answer after HINT TIMER MODERATE amount of time, then display Hint Thought Bubble 1.	 

Note. Case 1. Player answers first.
If the Player answers first, then

START Player animation movement RAISE HAND

Player confirms answer by selecting either yes or no button.
If the Player selects no, then question with answer options are re-displayed. If the Player selects Yes, the game progresses and the answer is evaluated.

START Player animation movement LOWER HAND

Note. Case 2. Competing character answers first.
START character animation movement RAISE HAND
Character confirms answer by selecting either yes or no button.
If the Character selects no, then question with answer options are re-displayed. If the Character selects Yes, the game progresses and the answer is evaluated.

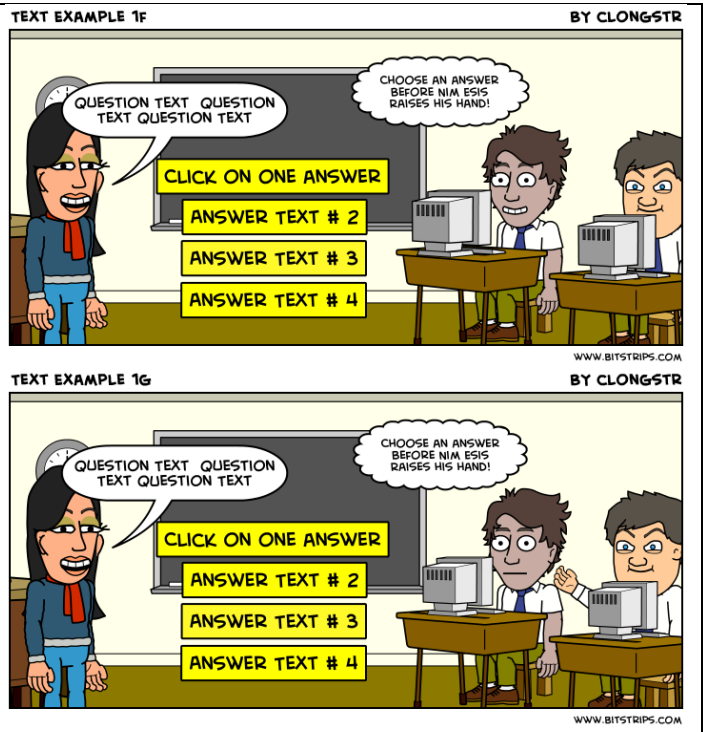
START character animation movement LOWER HAND

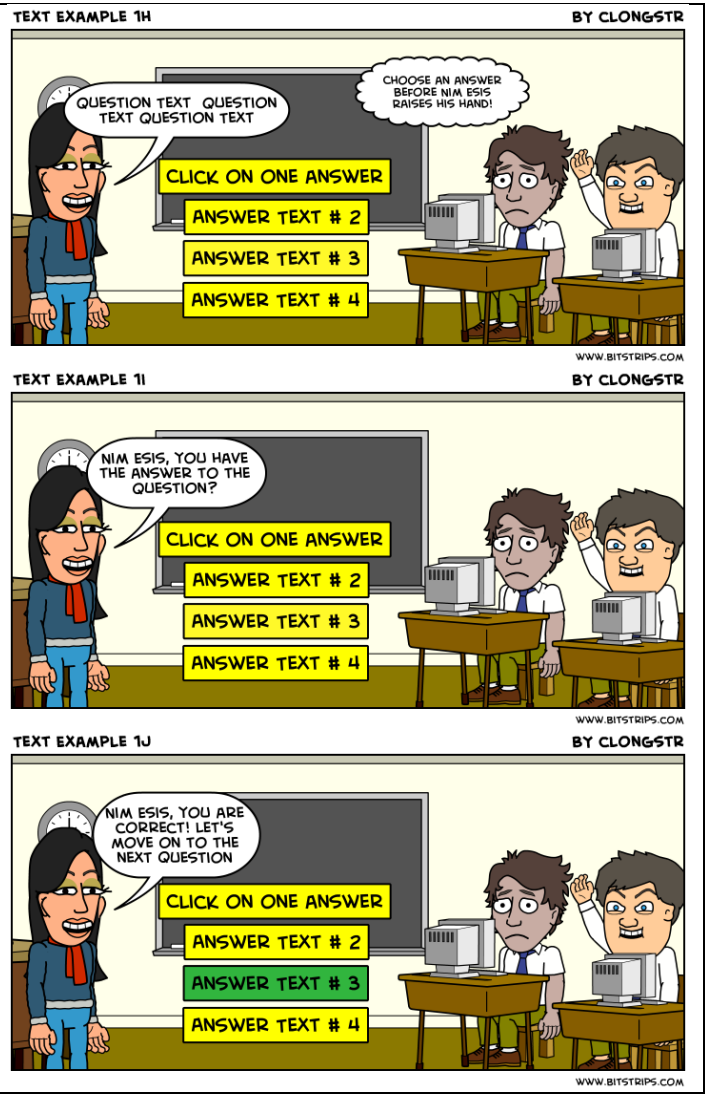
Note. Evaluation
If the Player or the Competing character answers correctly, then they are rewarded with 100 points.

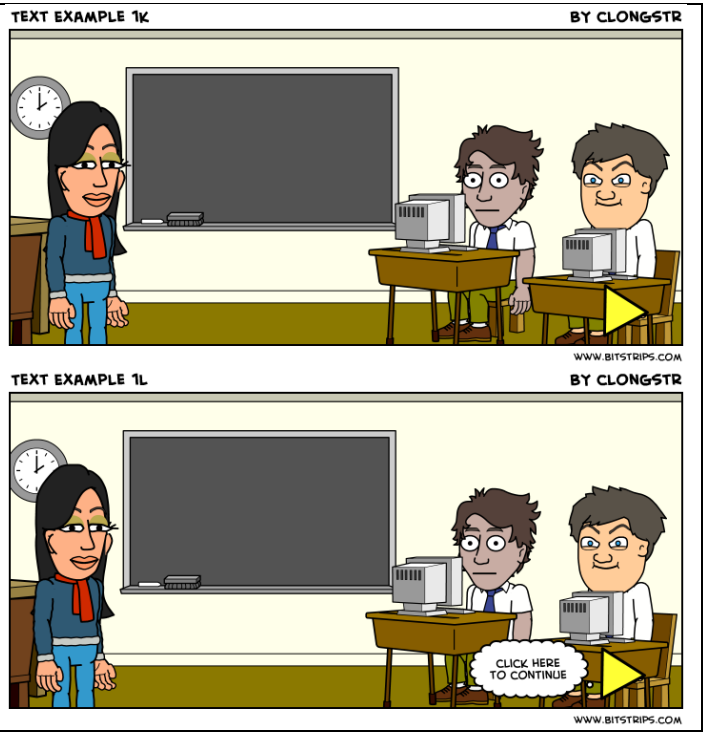
If the Player or the Competing character answers incorrectly, then they are penalized by deducting 100 points.

Note. Progress in the game (next question or the end of the quiz)

Note. When the answer has been confirmed, the right arrow is displayed, which allows the player to progress in the game.



	Player clicks the Right Arrow Button 1 to end the screen. If player has not selected the Right Arrow Button 1 after HINT TIMER MODERATE amount of time, then display Hint Thought Bubble 2. Player clicks the Right Arrow Button 1 to end the screen. **Note. At the end of the screen, the game needs to stop the audio and remove the visual setting from the display.**	

		
Timers	Not present.	
Game Play		
Behaviour state machine description: Current quiz element Transition event	**Note. Question Order** Question 1 Answer Q1.A3, follow-up question Q1F presented Question 2 Answer Q2.A3, follow-up question Q2F presented Question 3 Question 4	

condition output Next quiz element	Answer Q4.A3, follow-up question Q4.F1 presented Answer Q4.A5, follow-up question Q4.F2 presented	

4. Conclusions and Future Work

In this report, we have presented the first part of the Game Requirements for a Software Engineering Educational Game. The game is specified using a collection of new requirements templates. The templates have been inspired by existing requirements specification approaches: textual use case templates and graphic storyboarding; music files are embedded to specify the audio. The interactive behavior in the game is described (informally) as state machines. The requirements have been manually translated into XML game script, which can be loaded and run in the SimSYS Game Play Engine. We have found the templates straightforward to use, creating a flexible, modular specification that can be readily updated. Including the sample graphic UI storyboard cells and audio files enhances the textual specification to represent what-you-see-and-hear-is-what-you-get.

Two additional scenes are planned for Act I. Scene 3 is a job fair challenge, where the player will prepare a schedule in a Gantt chart for a project description. Scene 4 is a Boss challenge, where the player will select a team to staff their project. Act I concludes with a wrap-up of the Player's accomplishments. In Act II, the Player will run the development project with the schedule and team they prepared.

In the future there are several directions to pursue in the research. We plan to formalize the templates and automate the transformation from the templates into XML game scripts. A semi-automated, intelligent wizard is also planned to help create games across diverse domains; a flexible, re-usable approach to define, acquire, analyze, and report assessment and adaptation behavior is also being investigated. The emotional responses of the characters needs to be investigated; this is anticipated to provide interesting variability in the game play.

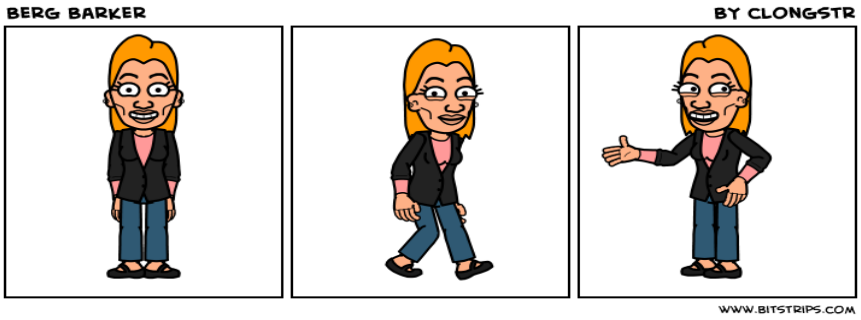
Acknowledgements

The graphics used in this specification have been developed using Bitstrips (www.bitstrips.com). The music files used are open source.

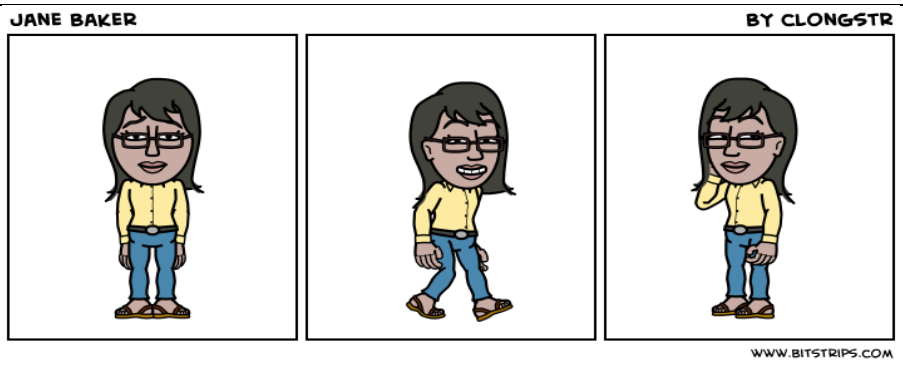
Special thanks to Microsoft Corporation, Software Engineering Innovation Foundation 2010 Award.

Appendix A. Employee Character Profiles

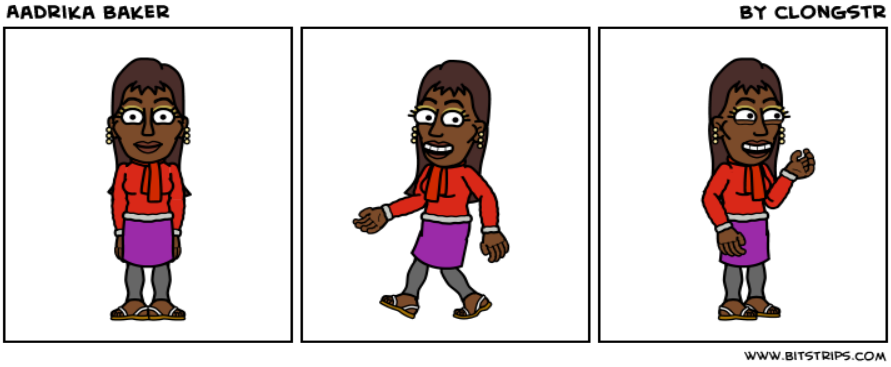
Berg Barker

Profile	
Name	Berg Barker
Title	Junior requirements engineer
Technical Skills	Petri nets, statecharts, UML use cases, IEEE 830
Years of Experience	4 requirements
Communication	Great
Leadership	Fair
Teamwork	Great
Demographics	Caucasian American, Female
Availability	M-F, 8 a.m. – 5 p.m.
Attendance	95%
Degrees	B.Sc. Computer Science, NorthEastern University
Resume Image	

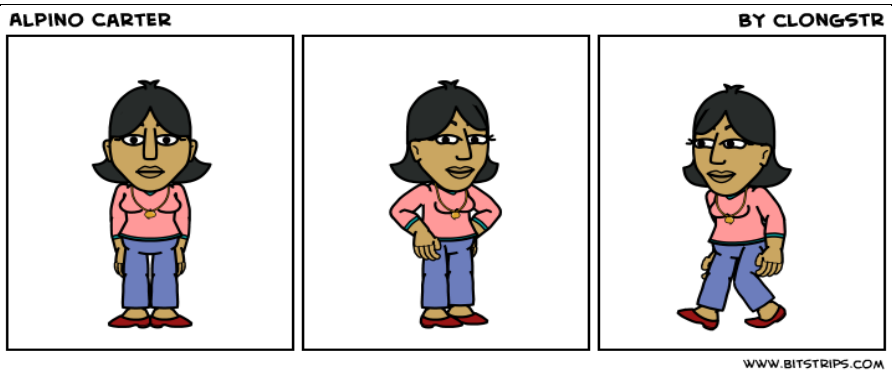
Jane Baker

Profile	
Name	Jane Baker
Title	Junior software developer
Technical Skills	C, C++, C#, Java
Years of Experience	3 programming
Communication	Great
Leadership	Great
Teamwork	Great
Demographics	Asian American, Female
Availability	M-F, 8 a.m. – 5 p.m.
Attendance	100%
Degrees	B.Sc. Computer Science, Texas A&M M.Sc. Computer Science, UT Austin
Resume Image	

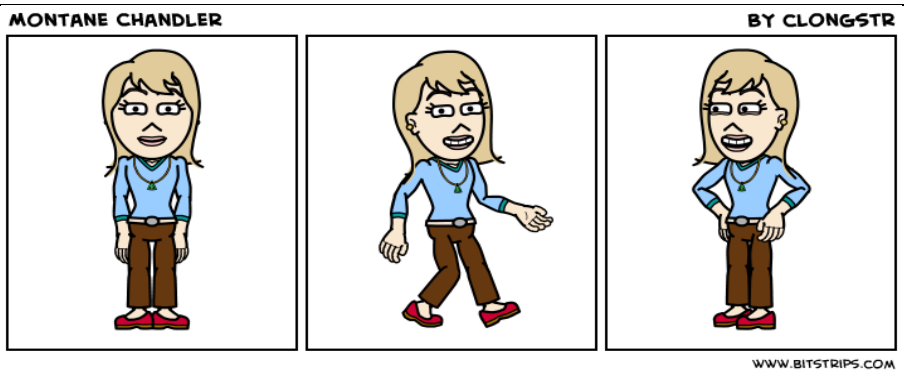
Aadrika Baker

Profile	
Name	Aadrika Baker
Title	Junior software designer
Technical Skills	C++, C#, Java, VB, .NET, J2EE
Years of Experience	3 design
Communication	Great
Leadership	Great
Teamwork	Great
Demographics	Asian Indian American Female
Availability	M-F, 8 a.m. – 3 p.m.
Attendance	95%
Degrees	B.Sc. Computer Science, Duke University
Resume Image	

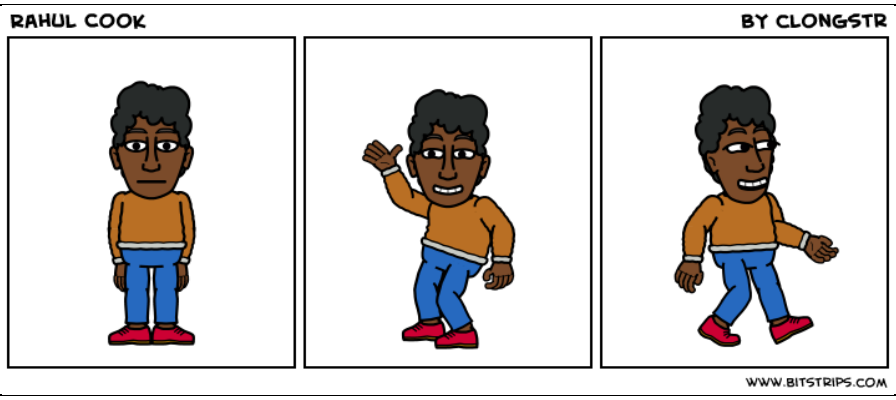
Alpino Carter

Profile	
Name	Alpino Carter
Title	Junior software designer
Technical Skills	C++, C#, Java, VB, .NET, J2EE, Spring, IEEE 1016, XP, Scrum
Years of Experience	4 design
Communication	Good
Leadership	Good
Teamwork	Fair
Demographics	Asian American Female
Availability	M-S, 8 a.m. – 5 p.m.
Attendance	95%
Degrees	M.Sc. Computer Science, Georgia Tech.
Resume Image	

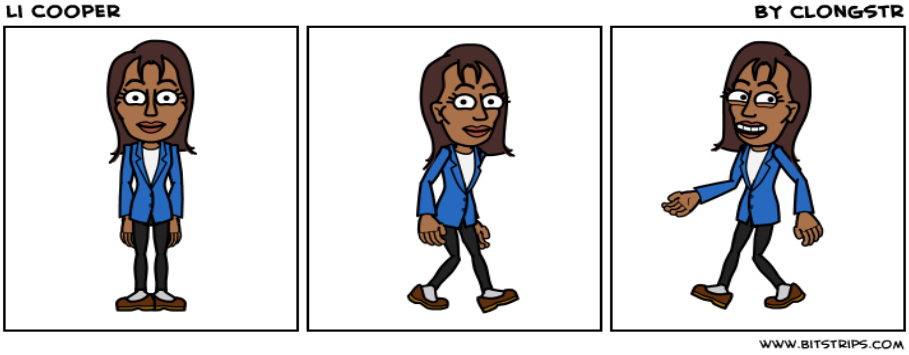
Montane Chandler

Profile	
Name	Montane Chandler
Title	Intermediate manager
Technical Skills	Unified Process, Scrum, IEEE 1220, PMI certified
Years of Experience	6 management
Communication	Excellent
Leadership	Excellent
Teamwork	Good
Demographics	Caucasian American, Female
Availability	M-F, 8 a.m. – 8 p.m.
Attendance	95%
Degrees	B.Sc. in Computer Science, USC M.B.A. USC
Resume Image	

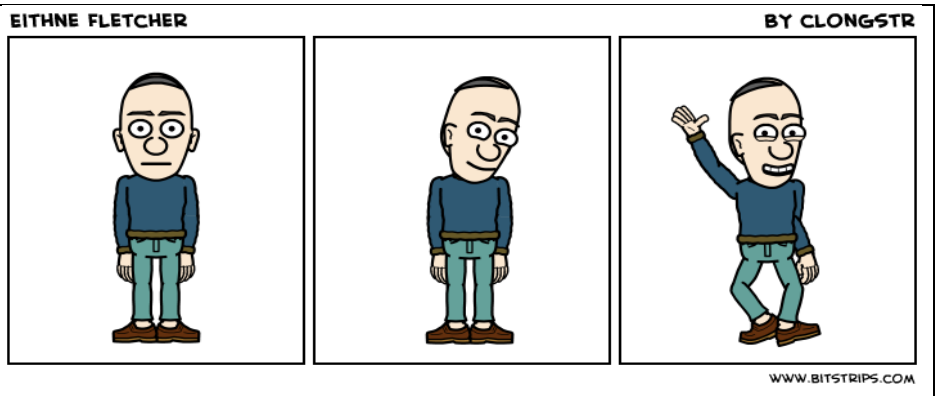
Rahul Cook

Profile	
Name	Rahul Cook
Title	Intermediate software developer
Technical Skills	C++, C#, Java, VB,.NET, PHP, Python, .NET, Android
Years of Experience	6 programming
Communication	Great
Leadership	Good
Teamwork	Good
Demographics	Asian American, Male
Availability	M-S, 8 a.m. – 8 p.m.
Attendance	95%
Degrees	B.Sc. in Computer Science, Stanford University Java certified developer
Resume Image	

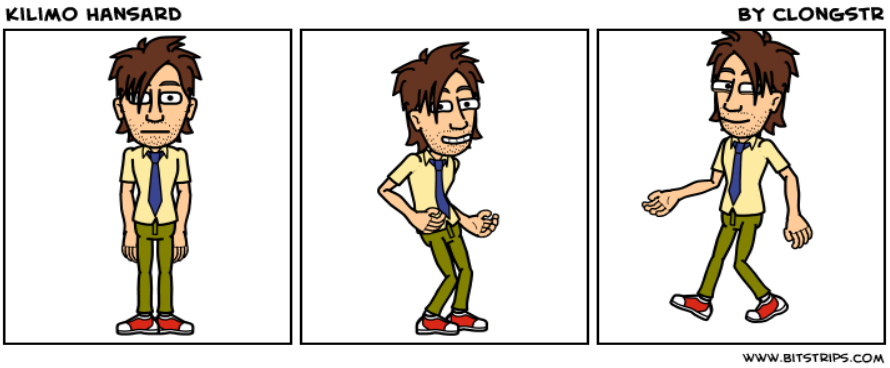
Li Cooper

Profile	
Name	Li Cooper
Title	Intermediate software designer
Technical Skills	C#, Java, VB,.NET, PHP, Python, Perl, Bash, Ruby, Powershell, IEEE 1016, Unified Process
Years of Experience	7 programming
Communication	Good
Leadership	Good
Teamwork	Good
Demographics	Native American, Female
Availability	M-F, 8 a.m. – 8 p.m.
Attendance	95%
Degrees	B.Sc. in Computer Science, UT Austin Ph.D. in Computer Science, Rice University
Resume Image	

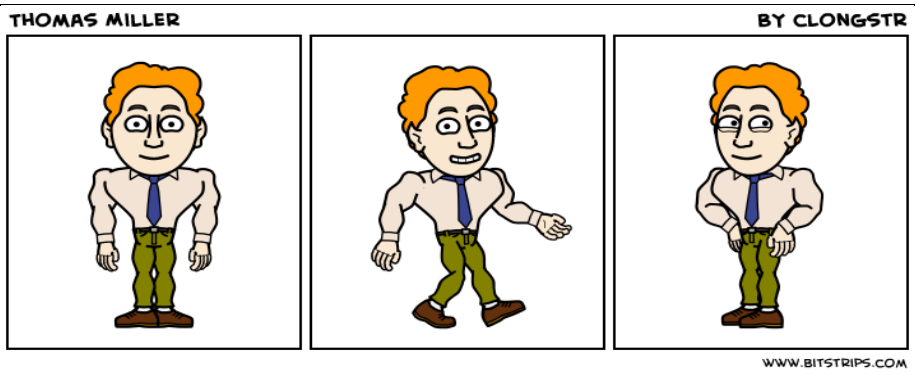
Eithne Fletcher

Profile	
Name	Eithne Fletcher
Title	Intermediate software requirements engineer
Technical Skills	C#, Java, VB,.NET, J2EE, Petri nets, statecharts, Z, UML use cases, IEEE 830, Storyboards, Scrum
Years of Experience	6 requirements
Communication	Excellent
Leadership	Excellent
Teamwork	Excellent
Demographics	Caucasian American, Male
Availability	M-F, 8 a.m. – 8 p.m.
Attendance	95%
Degrees	B.Sc. in Computer Science, UT Austin Ph.D. in Computer Science, Purdue University
Resume Image	

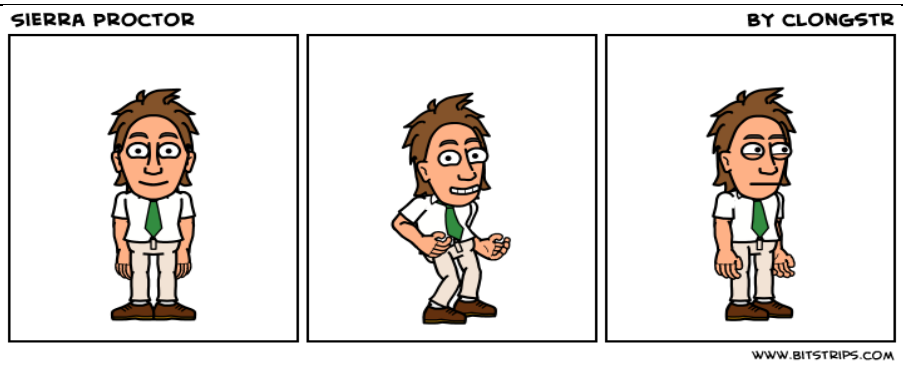
Kilimo Hansard

Profile	
Name	KilimoHansard
Title	Junior software developer
Technical Skills	Java, PHP, Python
Years of Experience	2 programming
Communication	Good
Leadership	Fair
Teamwork	Great
Demographics	Caucasian American, Male
Availability	M-F, 8 a.m. – 8 p.m.
Attendance	100%
Degrees	B.Sc. Computer Science, UT Austin
Resume Image	

Thomas Miller

Profile	
Name	Thomas Miller
Title	Senior software architect
Technical Skills	C, C++, C#, Java, IEEE 830, IEEE 1016, IEEE 1471, IEEE 1220
Years of Experience	12 architecture
Communication	Good
Leadership	Good
Teamwork	Good
Demographics	Caucasian American, Male
Availability	M-F, 8 a.m. – 8 p.m.
Attendance	95%
Degrees	B.Sc. in Computer Science, UC San Diego Ph.D. in Computer Science, Carnegie Mellon University
Resume Image	

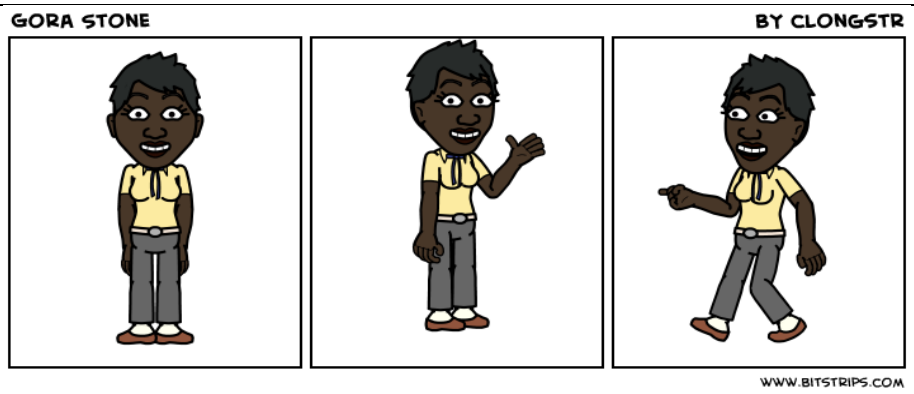
Sierra Proctor

Profile	
Name	Sierra Proctor
Title	Junior requirements engineer
Technical Skills	C++, C#, Java, J2EE, UML Use cases
Years of Experience	1 requirements
Communication	Great
Leadership	Great
Teamwork	Great
Demographics	Caucasian American, Male
Availability	M-F, 8 a.m. – 5 p.m.
Attendance	95%
Degrees	B.Sc. Computer Science, Northwestern University
Resume Image	

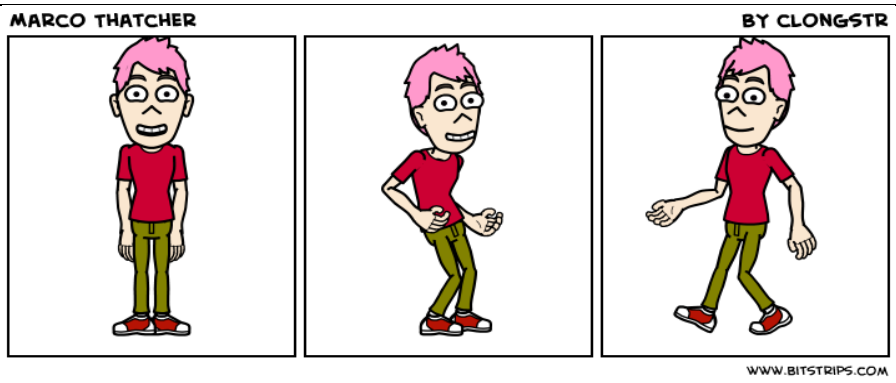
Ann Smith

Profile	
Name	Ann Smith
Title	Senior software developer ($45K-55K)
Technical Skills	C, C++, C#, Java, VB,.NET, J2EE, J2ME
Years of Experience	15 programming
Communication	Good
Leadership	Good
Teamwork	Good
Demographics	Caucasian American ,Female
Availability	M-S, 8 a.m. – 8 p.m.
Attendance	95%
Degrees	B.Sc. in Mathematics, Princeton University M.Sc. in Computer Science, Berkley University
Resume Image	

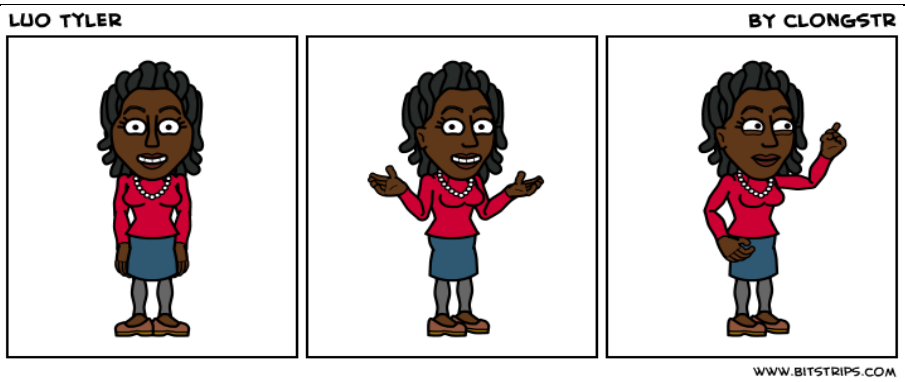
Gora Stone

Profile	
Name	Gora Stone
Title	Intermediate Business analyst
Technical Skills	BPMN, risk analysis, IEEE 1540
Years of Experience	6 business analyst
Communication	Excellent
Leadership	Excellent
Teamwork	Excellent
Demographics	African American, Female
Availability	M-F, 8 a.m. – 8 p.m.
Attendance	95%
Degrees	M.B.A. Harvard
Resume Image	

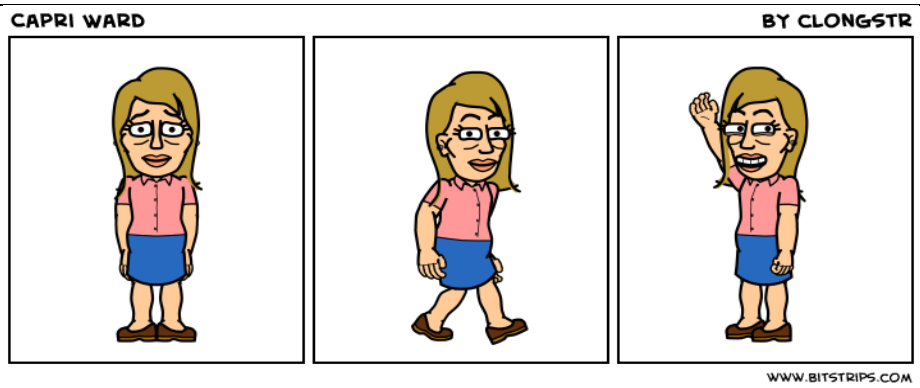
Marco Thatcher

Profile	
Name	Marco Thatcher
Title	Senior software tester
Technical Skills	C, C++, C#, Java, Perl, PHP, Python, Ruby, Tcl, Expect, Bash, IEEE 829
Years of Experience	12 years testing
Communication	Excellent
Leadership	Excellent
Teamwork	Excellent
Demographics	Caucasian American, Male
Availability	M-F, 8 a.m. – 8 p.m.
Attendance	90%
Degrees	B.Sc. in Mathematics, Cornell University M.Sc. in Computer Science, Yale University
Resume Image	

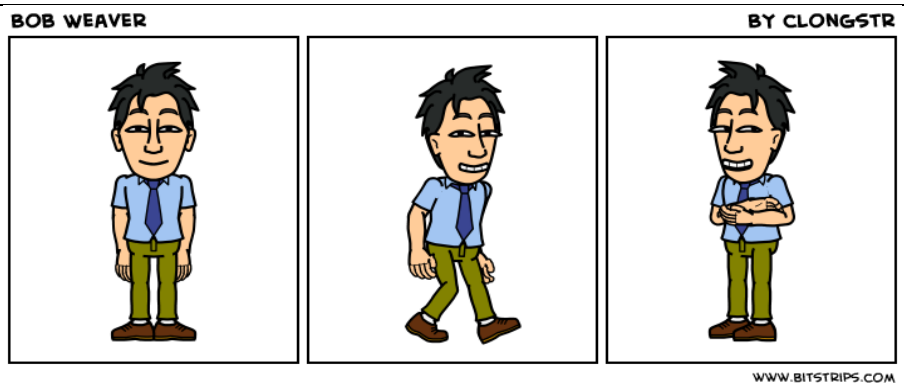
Luo Tyler

Profile	
Name	Luo Tyler
Title	Intermediate software tester
Technical Skills	C#, Java, VB,.NET, PHP, Python, Perl, Bash, Ruby, Powershell, IEEE 829
Years of Experience	7 programming
Communication	Good
Leadership	Good
Teamwork	Good
Demographics	African American, Female
Availability	M-F, 9 a.m. – 3 p.m.
Attendance	95%
Degrees	B.Sc. in Computer Science, Purdue University M.Sc. in Computer Science, Purdue University
Resume Image	

Capri Ward

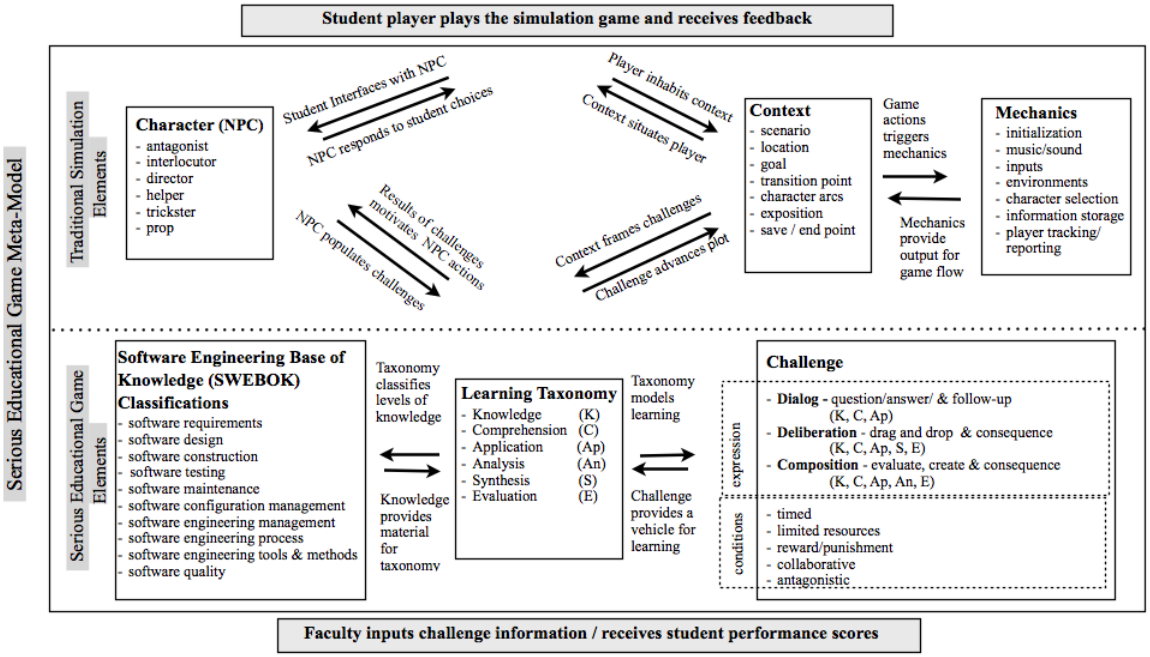
Profile	
Name	Capri Ward
Title	Senior requirements engineer
Technical Skills	C, C++, Struts, Spring, Petri nets, Statecharts, Z, B method, i*, Kaos, UML use cases, IEEE 830, Unified Process, Scrum
Years of Experience	8 requirements
Communication	Excellent
Leadership	Excellent
Teamwork	Excellent
Demographics	American, Caucasian, Female
Availability	M-F, 8 a.m. – 8 p.m.
Attendance	95%
Degrees	B.Sc. in Computer Science, UCLA M.Sc. in Computer Science, UCLA
Resume Image	

Bob Weaver

Profile	
Name	Bob Weaver
Title	Junior software tester
Technical Skills	C++, C#, Java, TCL, PHP
Years of Experience	3 testing
Communication	Great
Leadership	Great
Teamwork	Great
Demographics	Asian American, Male
Availability	M-F, 9 a.m. – 3 p.m.
Attendance	95%
Degrees	B.Sc. Computer Science, Columbia University
Resume Image	

Appendix B. SimSYS Game Foundations

B.1 Game Domain Meta-Model



B.1.1 External Entities

Student player – engages the simulation game and plays challenges, receives progress on self-progress.

Faculty/Developer – sets the parameters of the simulation, inputs the knowledge base and receives information on student progress.

B.1.2 Educational Game Elements

B.1.2.1 Challenge Class

Dialog Challenge –The player must answer a series of content-related questions that require her or him to then justify her/his answer in follow-up questions. This dialog challenge appeals to lower levels of a learning taxonomy such as Bloom's (i.e. knowledge, understanding, application of a concept/idea).

Deliberation Challenge – The player must pull apart a problem cluster provided in a director character's briefing. (extending learning into higher forms of learning such as synthesis, analysis and evaluation).

Composition Challenge – The player is required to analyze and evaluate a data set and, given a set of conditions and requirements, must create a select data group to solve a problem.

Conditions

Challenges may have conditions set upon them as a means to motivate the player; players can compete against time or a NPC character, with strategically limited resources or a combination of conditions.

- timed
- limited resources
- reward/punishment
- collaborative
- antagonistic

B.1.2.2 Learning Taxonomy

The simulation game leverages the challenges so that student players will demonstrate different degrees of critical thinking abilities. This simulation game uses a derivation of Bloom's Taxonomy (in order from simpler to more complex):

- Knowledge (K)
- Comprehension (C)
- Application (Ap)
- Analysis (An)
- Synthesis (S)
- Evaluation (E)

B.1.2.3 Body of Knowledge

The learning class represents the learning objectives. In software engineering, international standards determine competency. The S2004 guide organizes the Software Engineering Education Knowledge (SEEK) into three levels: knowledge areas; units; topics.

- software requirements
- software design
- software construction
- software testing
- software maintenance
- software configuration management
- software engineering management
- software engineering process
- software engineering tools & methods
- software quality

B.1.3 Traditional Game Components

B.1.3.1 Character

The Character class includes the Protagonist (student-player). There are a variety of non-player characters (NPC) who populate the game environment and engage the student players. Each NPC has an adjustable profile with the following attributes: role type. NPCs can have variable skill levels, attributes, and a range of visual representations that designers may choose from.

- antagonist
- interlocutor
- director
- helper
- trickster
- prop

B.1.3.2 Context

The Context Class provides the player/protagonist with the parameters of the simulation game world and narrative framework. The interplay between the player/protagonist within the plot context and the non-player characters facilitates understanding course concepts, and reinforces the effects of good and bad decision-making.

- scenario
- location
- goal
- transition point
- character arcs
- exposition
- save / end point

B.1.3.3 Mechanics

The Mechanics class provides the fundamental, low-level game resources such as characters, music, and graphics. It also supports maintaining the state of the game including the player's progress (where they are in the game) and their assessment (how well they are doing in the game, e.g. points). This class supports both the Context Class and the Challenge Class.

- initialization
- music/sound
- inputs
- environments
- character selection
- information storage
- player tracking/reporting

Appendix C Specification Templates

C.1 Game Template

// A Game is composed of 1 or more Acts; a Game controls Acts.

Table 19. Game Template

Identifier	*Game #*
Purpose	*Textual description, informal summary of the game.*
Uses Acts	*List of 1..i Act identifiers used in the game, i≥1.* *transitions, cuts* *type; duration (SLOW, MEDIUM, FAST)* *// See: http://en.wikipedia.org/wiki/Film_transition* *cut* *Straight Cut (default)* *Contrast Cut* *L Cut* *Form Cut* *Match Cut* *Parallel Editing Cut* *Jump Cut* *transition* *Fade in* *Fade out* *Dissolve* *Wipe* *Morph*
Learning objectives	*List of 0 or more learning objective descriptions in English.*
Style	*Style identifier* *Bitstrips cartoon* *Blade runner* *Note.* *The Styles indicates the look 'n feel for the game.*

	In the future, stylesheets will be defined, which will contain the default values for the prop declarations. This will simplify the declaration as default values do not need to be repeated for each prop. The declarations will be shorter; reduce the amount of manual data entry and reduce potential data entry errors.
Character	
Player	*Define the Player in the game (role, profile, rewards).* *Provide a list of one or more profile options (player may be allowed to select their avatar)* *Player has* *name* *meta-model type is protagonist* *initial location, pose, expression* *profile* *Resume image* *Degrees (subject, degree, school)* *Title (e.g., junior, intermediate, senior)* *Number of years of work experience* *Technical skills (e.g. requirements engineer, architect, tester, manager)* *Communication skills* *Leadership* *Teamwork* *Demographics* *Availability* *Attendance* *rewards* *points* *trophies* *certificates* *promotion/demotion* *behavior* *level of engagement (very positive, positive, neutral, negative, very negative)* *optional hint* *Notes.* *Profile can be updated during the game.*
Non-player	*Define the non-player characters in the game (role, profile, rewards)* *Non-player has*

	Name *Meta-model type (antagonist, interlocutor, director, constructor, trickster, or prop)* *initial location, pose, expression* *profile* *Resume image* *Degrees (subject, degree, school)* *Title (e.g., junior, intermediate, senior)* *Number of years of work experience* *Technical skills (e.g. requirements engineer, architect, tester, manager)* *Communication skills* *Leadership* *Teamwork* *Demographics* *Availability* *Attendance* *rewards* *points* *trophies* *certificates* *promotion/demotion* *behavior* *level of engagement (very positive, positive, neutral, negative, very negative)* *optional hint* *Notes.* *Rewards can be given or taken away.*
Behaviour state machine description: Current Transition event condition output	*Initial state for the Game* *Put state machine description here in English.* *Final state for the Game*

Next	

C.2 Act Template

// An Act is composed of 1 or more Scenes; an Act controls Scenes.

Table 20. Act Template

Identifier	*Act #* *The Act number is a unique, positive integer.*
Purpose	*Textual description, informal summary of the Act.*
Uses Scenes	*List of 1..j Scenes identifiers used in the Act, j≥1.* *Transitions, cuts* *type; duration (SLOW, MEDIUM, FAST)* *// See: http://en.wikipedia.org/wiki/Film_transition* *cut* *Straight Cut (default)* *Contrast Cut* *L Cut* *Form Cut* *Match Cut* *Parallel Editing Cut* *Jump Cut* *transition* *Fade in* *Fade out* *Dissolve* *Wipe* *Morph*
Learning objectives	*List of 0 or more learning objective descriptions in English.*
Behaviour state machine description: Current Transition event	*Initial state for Act #* *Put state machine description here in English.* *Final state for Act #*

condition output Next	

C.3 Scene Template

// A Scene is composed of 1 or more Screens; a Scene controls Screens.

Table 21. Scene Template

Identifier	*Scene #* *The Scene number is a unique, positive integer.*
Purpose	*Textual description providing an informal overview of the Scene.*
Learning objectives	*List of 0 or more learning objective descriptions in English.*
Uses Screens	*List of 1..k Screen identifiers used in this scene, k≥1.* *Transitions, cuts* *type; duration (SLOW, MEDIUM, FAST)* *// See: http://en.wikipedia.org/wiki/Film_transition* *cut* *Straight Cut (default)* *Contrast Cut* *L Cut* *Form Cut* *Match Cut* *Parallel Editing Cut* *Jump Cut* *transition* *Fade in* *Fade out* *Dissolve* *Wipe* *Morph*
Backdrop	*List of 1 or more backdrops.* *Each backdrop has* *name* *Notes.*
Behaviour	*Initial state for Scene #*

state machine description: Current Transition event condition output Next	*Put state machine description here in English.* *The Scene transitions from Screen to Screen* *Final state for Scene #*

C.4 Screen Template

// A Screen has game play elements (declarations of challenges, visual setting, and audio; behavior); a screen may be composed of zero or more Screens.

Table 22. Screen Template

Identifier	*Screen #* *The Screen number is a unique, positive integer.*	
Purpose	*Description of the screen in English.*	
Learning Objectives	*List of 0 or more learning objective descriptions in English.*	
Declarations, initialization		
Challenge	*List of 0 or more challenges.* *Challenge has* *Name* *Type, based on ontology: dialogue, deliberation, composition* *Currently, the dialogue challenge Quiz is defined.* *Notes. A challenge in a Screen is optional. For example, an introductory splash screen may not have a challenge.* *Transitions, cuts* *type; duration (SLOW, MEDIUM, FAST)* *// See: http://en.wikipedia.org/wiki/Film_transition* *cut* *Straight Cut (default)* *Contrast Cut* *L Cut* *Form Cut* *Match Cut* *Parallel Editing Cut* *Jump Cut*	

	transition *Fade in* *Fade out* *Dissolve* *Wipe* *Morph*	
Characters		
Player	*1 player character.* *Notes.* *This is a single player game.*	
Non-player characters	*List of 0 or more non-player characters.*	
Setting (visual)		
Props		
Generic Interaction	*0 or more generic interaction elements.* *Generic interactions are buttons, bubbles, information boxes, textboxes.* *Each generic interaction has* *Name* *Size* *Text* *Location* *Colour* *Optional hint* *Notes.*	
Education	*O or more education interaction elements.*	

interaction	*Education interactions are:* *blackboard* *whiteboard* *computer display* *paper flip-chart/easel* *clickers* *projector/screen* *each education interaction has* *name* *location* *size* *optional hint* *Notes.*	
Set Decorations	*Set decorations* *coffee cup* *office desk* *office chair* *office cubicle* *table* *speaker podium* *clouds* *each set decoration has* *name* *size* *location* *colour* *optional hint* *Notes.*	
Audio	*Music*	Embedded audio file(s)

	Sound effect *Voice over* *Notes.*	
Game Play		
Start of Screen	*Describe the initial state in English*	1 storyboard cell (figure)
Interactions (normal flow of events)	*Describe as a state machine in English* *Interactions are a state machine, with events, conditions, and output.* *Screen has transitions, cuts* *type; duration (SLOW, MEDIUM, FAST)* *// See: http://en.wikipedia.org/wiki/Film_transition* *cut* *Straight Cut* *Contrast Cut* *L Cut* *Form Cut* *Match Cut* *Parallel Editing Cut* *Jump Cut* *transition* *Fade in* *Fade out* *Dissolve* *Wipe* *Morph* *Screen element has animation:* *movement* *walk*	1 or more storyboard cells (figures)

<table>
<tr>
<td></td>
<td>
glide
sit
stand
talk
hand wave
hand shake
dance
effect
fade in
fade out
twinkle
shimmer

screen elements have optional hints that can be displayed, removed from the display

location defined by stage directions or absolute pixel co-ordinates.

Stage directions:
B backdrop
C center
U up
D down
R right
L left
O off

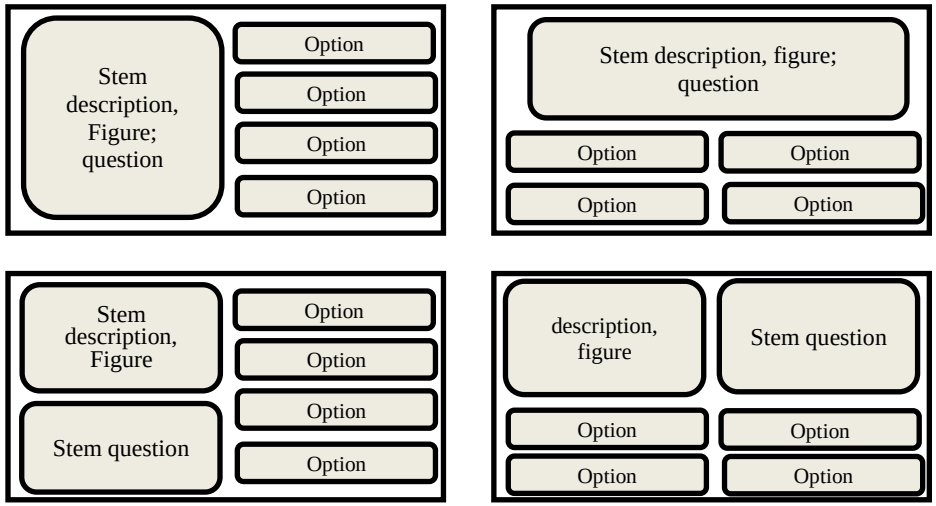
<table>
<tr><td colspan="7">Backdrop</td></tr>
<tr><td rowspan="6">OSR</td><td>UUR</td><td>UURC</td><td>UUC</td><td>UULC</td><td>UUL</td><td rowspan="6">OSL</td></tr>
<tr><td>UR</td><td>URC</td><td>UC</td><td>ULC</td><td>UL</td></tr>
<tr><td>R</td><td>RC</td><td>C</td><td>LC</td><td>L</td></tr>
<tr><td>DR</td><td>DRC</td><td>DC</td><td>DLC</td><td>DL</td></tr>
<tr><td>DDR</td><td>DDRC</td><td>DDC</td><td>DDLC</td><td>DDL</td></tr>
<tr><td colspan="5">Apron</td></tr>
</table>
</td>
<td></td>
</tr>
<tr>
<td>Alternate</td>
<td>Display game data</td>
<td>0 or more storyboard cell (figure)</td>
</tr>
</table>

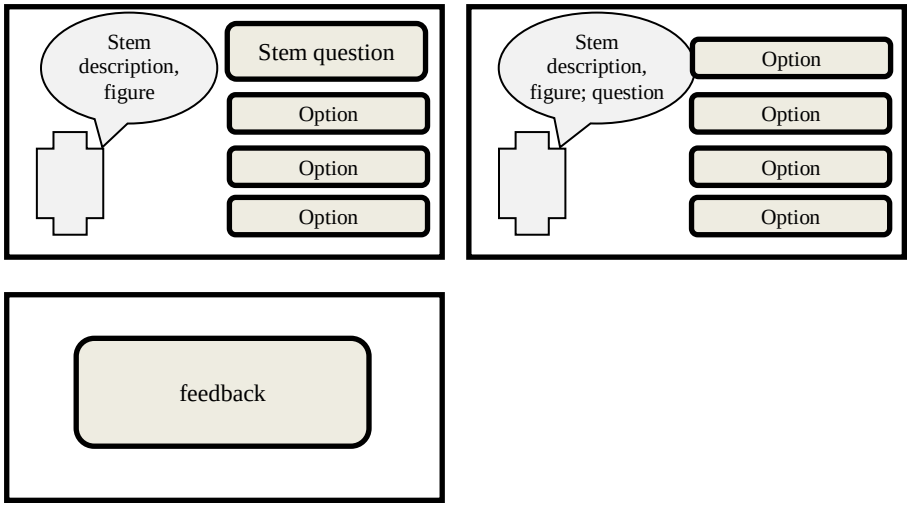
flow of events	*Character profile* *Character rewards* *The player can select a character and display the character's profile; the player can deselect a character and remove the profile from the display.* *The player can select a character and display the character's rewards; the player can deselect a character and remove the rewards from the display.* *Note. Use the alternate flow of events part of the template when the event is not tied to the normal progression of the game, in other words it is an asynchronous trigger to display game data.*	
End of Screen	*Describe final state(s) in English*	1 or more storyboard cell (figure)

C.5 Quiz Challenge Template

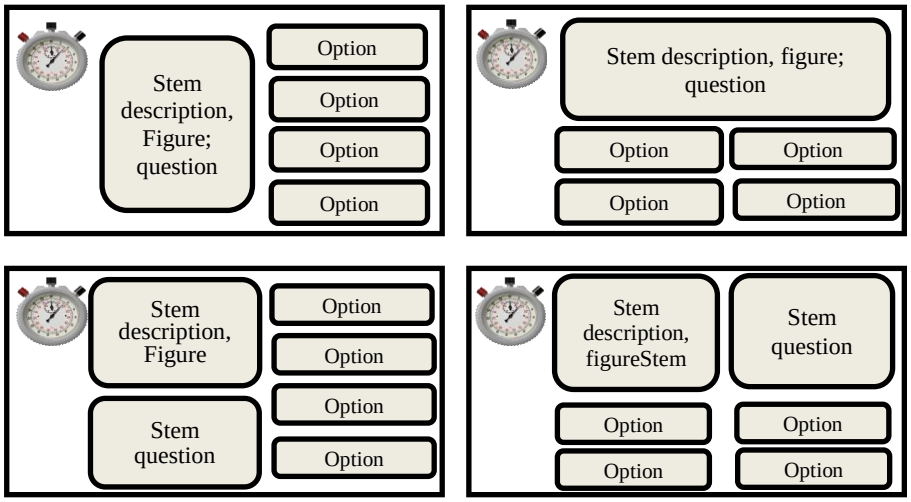
Table 23. Quiz Template

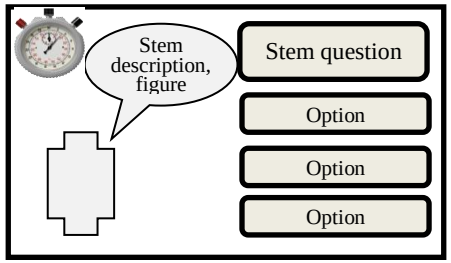
Identifier	*Quiz #* *The quiz number is a unique, positive integer.*
Purpose	*Textual description providing an informal overview of the Quiz.*
Learning Objectives	*List of 0 or more learning objective descriptions in English.*
Character presenting the quiz	*List of 0 or more characters presenting the quiz to the player.*
Competing Characters	*List of 0 or more NPCs the player is competing with in the quiz.* *NPC behavior?*

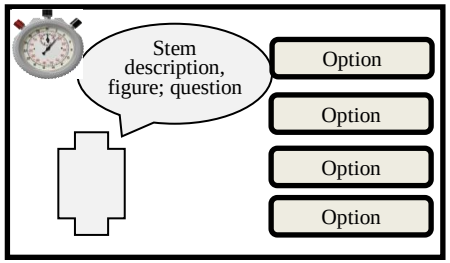
	Note. The quiz can be with or without a competing NPCs
Quiz Style	*Standardized layout, behaviour* *Type: multiple choice* *Layout (overall organization)* *Stem-description, figure* *Stem-question, figure* *Options, with hints* *Hints can be presented/hidden to/from the player upon request from the player or after a specified amount of time.* *Evaluation of each option (correct, incorrect, partially correct)* *Feedback* *// multiple choice quizzes* 

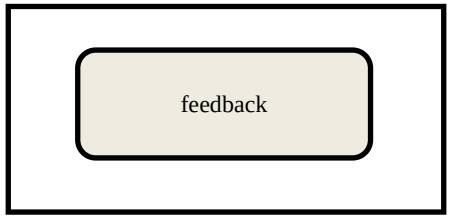


// timed multiple choice quizzes

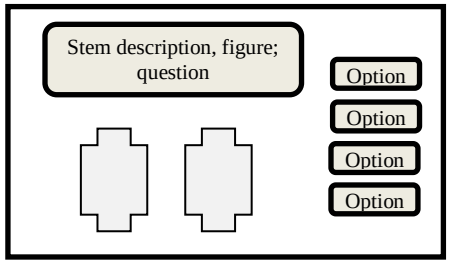


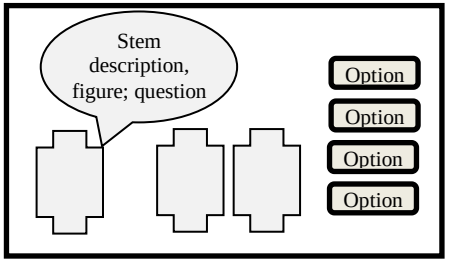


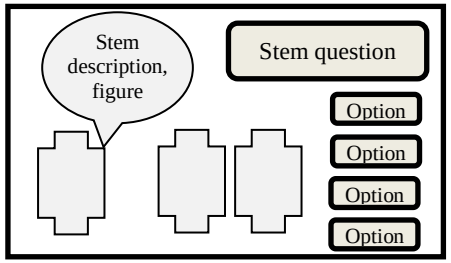


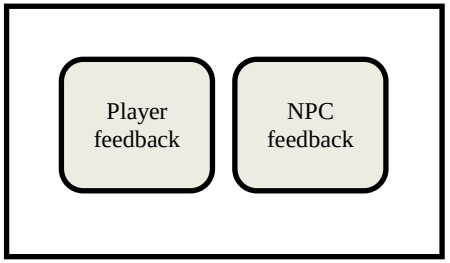


// competitive multiple choice quizzes

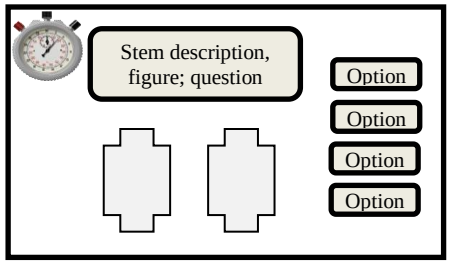


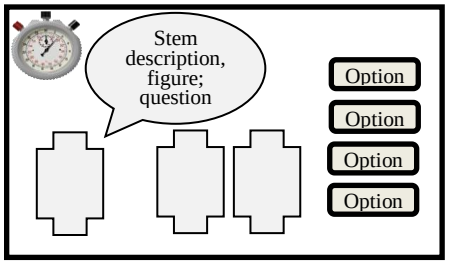


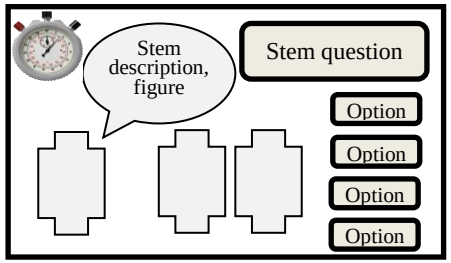


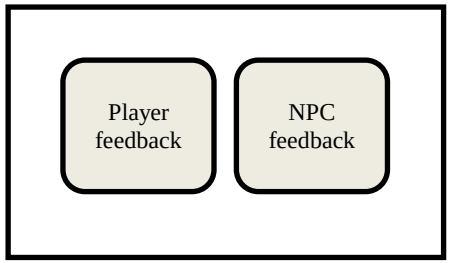


Timed, competitive multiple choice quizzes









Quiz elements	*List of l quiz elements, l ≥1.* *Note. A quiz element is a collection of <question, {(answer, evaluation, feedback)}>. One question can have one or more answers. An answer can be a correct answer, incorrect answer, or moderately correct answer.* *A moderately correct answer can lead to asking the player an additional, follow-up question for further reflection.*
Reward scheme	*Reward/penalty for a correctly answered or incorrectly answered question.* *Reward/penalty for a correctly answered or incorrectly answered follow-up question*
Declarations, initialization	
Quiz interaction props	*Declare how all the questions, answers, feedback are displayed; declare how the player selects an answer.* *Stem* *Options, with Hint* *Feedback* *Question text can be displayed on a generic interaction prop or an education interaction prop.* *Answer text can be displayed on a generic interaction prop or an education interaction prop.* *Answer selection can be displayed on a generic interaction prop or an education interaction prop.* *Feedback can be displayed on a generic interaction prop or an education interaction prop.* *They are defined for use throughout the quiz. For example, if an answer button is declared, then the button is used for every answer.* *Generic interactions are buttons, bubbles, information boxes, textboxes.* *Each generic interaction has* *Name* *Size* *Text* *Location* *Colour*

	Education interactions are: *blackboard* *whiteboard* *computer display* *paper flip-chart/easel* *clickers* *projector/screen* *each education interaction has* *name* *location* *size* *text*
Timers	*List of 0 or more timers used in the quiz.* *Note. This is needed for timed quizzes (with or without competing NPCs)*
Game Play	
Behaviour state machine description: Current Transition event condition output Next	*Describe the item order.* *The sequencing of items is explicitly specified.* *For example, at the end of item 1, the following can be specified:* *If condition 1, then proceed with item 2.* *If condition 2, then proceed with item 3.* *…* *If condition n, then proceed to item x.*