

**The Case of the killer Robot**

* Fictitious case written by Richard Epstein
* presented as a series of news reports and interviews.

These reports can be found at:

* <http://www.researchgate.net/profile/> Richard\_Epstein5/publication/242362 942\_The\_case\_of\_the\_killer\_robot/links

/5523ef280cf2c815e073e5a0.pdf



**The Case of the killer Robot**

#### Preparation:

The tasks you will need to complete (only a guide):

* Read all the documents in The Case of the killer Robot
* Attend the support sessions to aid your analysis of the scenario
* Meet regularly outside of class to discuss the topic
* Carry out critical analysis into Computing and IT within a Legal, Social, Ethical and Professional Framework
* Write a presentation on PowerPoint slides (at least 3 slides *pp*)
* Make sure each member of the group participates
* Practice performing your presentation
* Give your Presentation (only to me)
* Ethics addresses questions about morality, right and wrong, doing the right thing



**ETHICS**

* Most professions have an code of conduct *(set of rules outlining the responsibilities or proper practices)* for an individual or organisation
* Most professionals agree that building and maintaining a comprehensive ethics and compliance program and strong ethical culture are essential to good business.

Copyright © 2015 Pearson Education, Inc.

Ethics asks us … to live

“mindfully”

Have you been an ethical

team member today?



What ethical decisions have you made today?



**Ethical questions**

#### Business Ethics

* + What level of gifts and hospitality is appropriate?
  + Hotpoint Dryers. Do not leave them unattended?
  + VW Emissions scandal?

#### Research ethics:

* + Stem cell research falsified – Hwang Woo-Suk, South Korea
  + Should people be allowed to clone themselves?
  + Should we trust robotic cars to share our road?
* How do legal and ethical considerations interact?



**Duty to the Profession**

* Share knowledge and understanding of IT
* Support inclusion of every sector of society
* Encourage and support others professional development
* Membership of another professional body does not negate the BCS Code of Conduct
* Only make statements supporting or not the BCS through any public medium unless authorized by BCS



**Issues raised in the case study**

* The reason CKR has been chosen as a unifying theme to run through this unit is that

1. it raises many pertinent, but potentially dull, issues in a very entertaining form.
2. Hopefully, by the time you complete this unit, you will agree that Epstein is to be applauded for this achievement.

* In subsequent topics we shall be looking into these general categories in more detail, and you will be researching specific ideas and methodologies in even more detail. Eventually we shall amass sufficient information to attempt an answer to the question of culpability in this case.

## Issues raised in the case study



of various issues attempt can be imposed to classify them into four broad categories, as follows -

#### Requirements Analysis and Specification

* + Was the solution proposed the correct one for the customer?
  + Did the specified system produce the expected solution?

#### System Design and Development

* + Was the design method employed appropriate for the specified system?
  + Did the development methods and team suit the design?

## Issues raised in the case study



of various issues attempt can be imposed to classify them into four broad categories, as follows -

#### System Testing and Usability

* + Was the system thoroughly tested?
  + Were the users of the system considered/involved?

#### Ethics and Culpability

* + Did everybody concern act with the best of motives?
  + Who should be held responsible for the fatality?



**Computer Science ethical issues?**

* Computer crime, responsibility for computer failure
* Privacy, sharing data
* Cyber bullying
* Hacking, unauthorized access, viruses, sabotage
* Face recognition software on CCTV
* License plate reading can track vehicles
* Log files and their use
* Giving back to open source, share enhancements



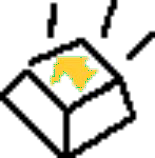
**Killer Robot: The news 1**

* A robot operator, Bart Matthews, was killed by his robot, Robbie CX30, while at work.
* A programmer, Randy Samuels, who wrote the faulty code was indicted for manslaughter.
* Silicon Techchronics is Randy’s employer.



**What really happened?**

* + The robot malfunctioned & crushed its operator to death



* + Blood all over
  + Decapitated head
  + Num-lock key light not on
  + Numeric keypad bloody



**Who is at fault?**

* + The programmer who made a careless mistake.
  + The designer?
  + The robotics division?
  + The company?
  + The operator of the robot?
  + The robot?



**CASE Summary**

* Bart Matthews, a robot operator at Cybernetics, Inc., has been killed by an out-of-control robot named Robbie (CX30).
* creator of the robot, Silicon Technologies, is also in a tight financial position and had hoped that the robot would put the company back on its feet.

What are the causes?

What is the relevance of this case to Computer Science and Software Engineering?

*OR*

What are the possible solutions?



**What are the causes?**

* Improper methodology was used in developing the software.
* Testing of the software was faked.
* The company pressured Robot’s creators to by-pass testing.
* Part of the software used in the robot was stolen
* from another vendor’s application.
* programmer did not understand or know the code which he used.
* Security measures used were illegal, and therefore all information gathered regarding the case might not be permissible in court.
* The project leader did not understand or use proper design methodologies.
* The end-user interface was designed improperly.



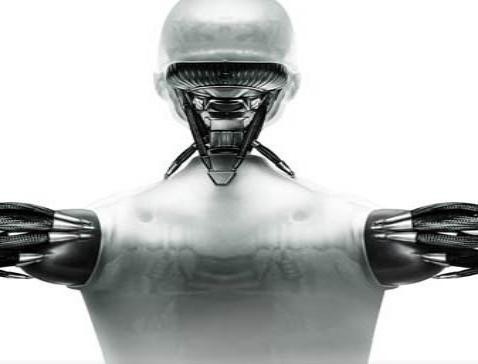
**Relevance of this case to Computer Science and Software Engineering**

* Follow the proper guidelines in creating a project.
* Don't rush everything.
* Work according to the plan.
* In a project like this. Budget or money is a must.
* Analyze Everything. As much as possible, analyze every possible errors that can occur in future.



## When software engineering and

**computer ethics collide**





**Characters**

* **Alex Allendale**, Attorney, hired to defend Randy Samuels.
* **Jan Anderson**, Former Programmer-Analyst at Silicon Techtronics. She opposed the use of the waterfall model on the robot project and was fired for her honesty.
* **Turina Babbage**, President of the Association for Computing Machinery (ACM). She announces an investigation by the ACM into violations of the ACM Code of Ethics by employees at Silicon Techtronics.
* **Robert Franklin**, Reporter for the Silicon-Valley Sentinel Observer. He interviewed Professor Harry Yoder in order to see how an ethicist would view the developments in the killer robot case. The interview was published in the Sentinel-Observer's Sunday magazine.
* **Horace Gritty**, Professor of Computer Science and Related Concerns at Silicon Valley University. He sees poor interface design as a primary cause of the killer robot tragedy.



# Characters

* **Sandra Henderson**, Graduate Student at Silicon Valley University. She assisted in the investigation into quality assurance procedures at Silicon Valley University.
* **Ray Johnson**, Robotics Division Chief at Silicon Techtronics. The Robotics Division needed a successful robot.
* **Martha**, Anonymous newspaper source. She is the insider at Silicon Techtronics who gave the Silicon Valley Sentinel-Observer information about the group dynamics on the Robbie CX30 robot project.
* **Bart Matthews**, Robot Operator. A faulty computer program caused a Robbie CX30 robot to strike him dead.
* **Roberta Matthews**, Widow of Bart Matthews.
* **Jane McMurdock**, Prosecuting Attorney for the City of Silicon Valley. She brought the manslaughter charges against Randy Samuels.
* **Mabel Muckraker**, Reporter for the Silicon Valley Sentinel-Observer. She was put on the killer robot story because of her reputation as an effective investigative reporter.



# Characters

* **Bill Park**, Professor of Physics at Silicon Valley University. He confirmed that Randy Samuels misinterpreted the robot dynamics equations.
* **Randy Samuels**, Programmer. He wrote the program code that caused the Robbie CX30 robot to oscillate wildly, killing the robot operator, Bart Matthews.
* **Sam Reynolds**, CX30 Project Manager. Ray Johnson was his immediate boss. His background was in data processing, but he was put in charge of the Robbie CX30 project, much to Ray Johnson's chagrin. He was committed to the waterfall model of software development.
* **Robbie CX30**, The Robot. Robbie never had an unkind thought about anyone, yet he turned into a savage killer.
* **Wesley Silber**, Professor of Software Engineering at Silicon Valley University. He conducted a review of software quality assurance procedures at Silicon Techtronics.
* **Sharon Skinner**, Professor of Software Psychology at Silicon Valley University. She saw Randy Samuels as a task-oriented person who was overly sensitive about criticism.



**Characters**

* **Valerie Thomas**, Attorney, hired by Sam Reynolds.
* **Michael Waterson**, President and CEO of Silicon Techtronics. Placed Sam Reynolds in charge of Robbie CX30 project as a cost-saving measure. He contributed generously to Jane McMurdock's re-election campaign. He hired Dr. Silber to conduct investigation into software quality assurance at Silicon Techtronics.
* **Ruth Witherspoon**, Programmer-Analyst and spokesperson for the "Justice for Randy Samuels" committee. She defends Randy Samuels on the grounds that Silicon Techtronics was legally obligated to deliver a safe robot.
* **Max Worthington**, Chief Security Officer for Silicon Techtronics. He monitored electronic mail communications among the employees and thus exposed Cindy Yardley.
* **Cindy Yardley**, Silicon Techtronics employee and software tester. She admitted to faking software tests in order to save the jobs of her co-workers.
* **Harry Yoder**, Samuel Southerland Professor of Computer Technology and Ethics. He examines the tension between individual and corporate responsibilities in interview published by Sentinel-Observer's Sunday magazine.



**Summary of the Article1**

The following provides an overview of the topics and issues addressed by each of the "Killer Robot" articles.

Article #1 - Silicon Valley Programmer Indicted for Manslaughter

* This article sets the stage and asks the basic question, “Should programmers be held responsible for the effects of errors in their programs?”
* Maybe this is why most software license agreements include statements such as the following:
* In no event shall XXXXX or its suppliers be liable for any damages whatsoever (including, without limitation, damages for loss of profits, business interruption, loss of information, or other pecuniary loss) arising out of the use of or inability to use this XXXXX product, even if XXXXX has been advised of the possibility of such damages. Because some states do not allow the exclusion or limitation of liability for consequential or incidental damages, the above limitation may not apply to you.

discusses programs/software in context to programmer psychology, team dynamics, user interfaces, software process models, software testing, the nature of requirements, software theft, and privacy.

A major consideration is "when is the software good enough?"

students are introduced to factors within software company that also contributed to accident.

it shows that a software development as a social process.

It is hoped that students will begin to realize the complexity of the task of building real-world software and to see some of the ethical issues intertwined in that

Copyright © 2015 Pearson Education, Inc.

complexity

Ethical questions to raise?

Explain why it is not a good idea to add new programmers to a project that is already late?

Make a list of arguments for and against Ray Johnson's Ivory Snow theory. Are your points valid regardless of the kind of system being considered?

Are there systems for which the Ivory Snow theory makes sense? Are there systems for which it does not make sense?



**Summary of the Article 2**

Article #2 - Developers of ‘Killer Robot’ Worked Under Enormous Stress

* The pressure of deadlines.
  + Get the product out the door by a given date or “heads will roll.”
  + Is this why so many software products have version upgrades that consist of nothing more than bug fixes?
* This article also introduced the “Ivory Snow Theory” of software design and development.
  + if the product is 99.44% pure, is there a no reason to try to improve it.
  + “Perfect software is an oxymoron.”
  + there is no such thing as a bug free program, only programs where the bugs have not yet been found. That is,

"how good is good enough?"

Article #3 - ‘Killer Robot’ Programmer was Prima Donna, Co-Workers Claim

**Team Dynamics**

A good programming team requires a mixture of personality types,

* who is interaction-oriented,
* who derives lot of satisfaction from working with others,
* someone who can help keep the peace and
* keep things moving in a positive direction.

However,

Most programmers are task-oriented,

could it be a problem for a team in which everyone is task-oriented?

Randy Samuels is a "prima donna",

* found it very difficult to accept criticism,
* or more accurately, to accept his own fallibility.
* task-oriented personality, bordering on self-oriented.
* likes to get things done, but his ego is heavily involved in his work.

Whether you like it or not BUT

In the programming world this is considered a no-no.

Samuels team consisted of a :

* dozen analysts, programmers and software testers.

*This does not include the extra twenty programmers who were later hired but who never became actively involved in the development of the robotics software.*

* Although individual team members had definite specialties, almost all were involved in the entire software process from beginning to end.
* Democratic team (which is good), but also very ambitious, very talented and very opinionated individualists

Unfortunately;

“Randy (Samuels) was just the worst of the lot”

In one incident, (code-review meeting / quality assurance)

“One of the readers mentioned that Samuels had used a very inefficient algorithm (program) for achieving a certain result and Samuels "turned beet red."

Samuels yelled a stream of obscenities before he left the meeting and never returned.

“Team sent him a memo about faster algorithm and he eventually did use the more efficient algorithm in his module,"

The software module in the quality-assurance incident was the very one found to be at fault in the death of robot operator Bart Matthews.

However, this co-worker was quick to point out that the efficiency of the algorithm was not an issue in the malfunctioning of the robot:

Randy’s a character:

* very difficult for people to communicate their concerns to him.
* took everything very personally.
* graduated tops in his class at college and later graduated with honors in software engineering from a first-class University. definitely very bright."

"Randy had this big computer-generated banner on his wall.

It said, 'You give me the specification and I'll give you the computation.'

* That demonstrates his arrogance
* had little patience for developing and checking the specifications.
* loved the problem-solving aspect, the programming itself.

Ego-less programming

software product belongs to team and not to individual programmers.

Ideas is:

* be open to criticism
* be less attached to one's work.

Code reviews are certainly consistent with this overall philosophy."

Another aspect of Samuels' personality was his helpfulness:

* hated meetings, but he was pretty good one on one.
* always eager to help.

One co-worker said;

“I remember one time when I ran into a serious roadblock and instead of just pointing me in the right direction, he took over the problem and solved it himself. He spent nearly five entire days on my problem Of

course, in retrospect, it might have been better for poor Mr. Matthews if Randy had stuck to his own business."

Article #3 - ‘Killer Robot’ Programmer was Prima Donna, Co- Workers Claim

This article considers the psychology of programmers. There are two basic types:

(⑴)most common is task-oriented = gets things done but ego is heavily involved; and

(⑵)interaction-oriented = derives satisfaction from working with others.

Primary issue: major programming projects need to be a team, not individual, effort.

Article #4 - ‘Killer Robot’ Project Mired in Controversy Right from Start

* Two software development philosophies:
* waterfall model: Follows definite stages of development with little chance for going back and changing earlier decisions. Potential users interact with the system very late in the process.
* prototyping model: A working model is produced early in the project for the purpose of refining the specifications throughout the process. Potential users interact early and often throughout the project.
* The waterfall model works quite well with data processing types of applications that require very little user interface. But where there is a high level of user interface, prototyping is the preferred approach.

Article #5 - Silicon Techtronics Promised to Deliver a Safe Robot

* This article introduces the concepts of a “requirements document” (a contract between the supplier and the customer giving the details of the functionality of a product as well as other agreements).
* One aspect of this requirements document is the training of operators. In points out the importance of quality operator training. How much training is necessary? Is training necessary? Without proper training, how will users really know what the software can or will do, or the best way to do it??

Article #6 - The ‘Killer Robot’ Interface (Journal Article)

* This article considers interface design and looks at Shneiderman’s “Eight Golden Rules” of user interfaces. These rules include,
* Strive for consistency.
* Enable frequent users to use shortcuts.
* Offer informative feedback.
* Design dialogues to yield closure.
* Offer simple error handling.
* Permit easy reversal of actions.
* Support internal locus of control (i.e. give the user a feeling of control, not being controlled).
* Reduce short-term memory load (Miller’s Law = short-term memory is limited to 7 discrete pieces of information).

Article #7 - Software Engineer Challenges Authenticity of ‘Killer Robot’ Software Tests

* This article considers the issue of quality testing. Testing a program is very difficult because the data used must check all possible outcomes. One quote in this article notes the difficulty of software testing, “Large computer programs are arguably the most complex artifacts ever fashioned by the human mind.”
* Several types of testing are mentioned, including:
* dynamic testing = testing code by running it against test data
* static testing = programs that analyze the code for suspicious patterns that might indicate a problem
* unit testing = testing individual components independent of the rest of the program
* integration testing = testing the combined components when they interface with each other
* black box testing = checking the input/output behaviors of a component of code
* white box testing = checks all possible paths through a unit of code (all statements of the program must be executed at least once)



**Summary of the Article 8**

Article #8 - Silicon Techtronics Employee Admits Faking Software Tests

* This article addresses the issue of electronic surveillance of employees and, in particular, the privacy issues related to electronic mail.
* As secondary issue brought up in this article is the ACM (Association of Computing Machinery) code of ethics. This includes:
* contribute to society and human well-being
* avoid harm to others
* be honest and trustworthy (i.e. integrity)
* give proper credit for intellectual property
* access communication resources only when authorized
* respect the privacy of others
* honor confidentiality.

Article #9 - A Conversation with Dr. Harry Yoder (Magazine Article)

* An overview of the ethical issues involved in this case. More specifically:
* The Jewish tradition concerning the shedding of innocent blood (i.e. the taking on of responsibility [not guilt] for the actions of the members of an organization or community).
* The five ethical tests of decision making:
* Mom Test: Would you tell your mom what you have done?
* TV Test: Would you broadcast what you have done on national TV?
* Smell Test: Does it have a bad smell?
* Other Person's Shoes Test: Would you like this done to you?
* Market Test: Would these actions be a good sales pitch?



Analysis

/ Interpertation



**Robbie CX 10 & 20 were experimental.**

* The robotics division chief, Johnson, put pressure on the project manager Reynolds to finish the project by January.
* Johnson hired 20 new programmers in June against Reynolds' will, by shifting resources. Robbie CX 30 had to succeed.
* Johnson only knew manufacturing hardware. One cannot speed software coding by adding more programmers.
* Robbie CX 30 was a step ahead in sophistication.
* Johnson: Perfect software is an oxymoron.



**Programmer personality**

* Randy’s home page: freedom for programmers
* A hacker type
* Enormous stress
* Programmer was a prima donna: could not accept criticism or his own fallibility
* Helpful but arrogant
* Under pressure to finish project on time



**A good programming team**

* + Needs to have a whole array of personalities types including:
  + An interaction-oriented person
  + Someone who keeps the peace and helps move things in a positive direction
  + Task-oriented people, but not all
  + A democratic team
  + Egoless programming



**Team dynamics**

* + The project was controversial from the beginning
  + Jan Anderson, a programmer, was fired after she attacked project manager Reynolds on his decision to use the waterfall methodology versus the prototype methodology.
  + Reynolds was replacing a dead project manager as a cost saving measure instead of hiring someone from outside the company. He never worked with robotics before.



**More explanations**

* + The earlier Robbie models processed transactions. Robbie CX30 was to interact with its operator, therefore the interface cannot be designed as an afterthought. Thus, a prototyping model is preferable where the users can use a prototype robot while in the design stage.



**Software development life-cycle Water fall methodology**

* + - Requirements
    - Analysis
    - Design
    - Coding
    - Testing
    - Implementation
    - Maintenance
    - No standard practices



**Quality control or quality assurance**

* + Dynamic testing
  + Static testing
  + Unit testing: Black and white box testing
  + Integration testing
  + Prof Silber attested that test results were inconsistent with actual killer code test results



**When is the software good enough?**

* + Errors can happen in any stage
  + Design stage: 2 alternatives: Waterfall & prototyping
  + Testing
  + User interface
  + The complexity of the task of building real-world software
  + Enormous stress

Worker-friendly: Chip Creek facility. Prevent repetitive strain injuries.



**The company-Sili-Tech Two environments:**

Special training for employees.

Well designed workstations. Frequent breaks.

Worker-unfriendly: Silicon Valley plant.

Workers criticized for novelties.

No exercises or training encouraged.

RSI frequent

More compensation claims



**Law suits**

* + - The wife of the murdered operator.
    - The indicted programmer himself.
    - More possible indictments.



**Why was Randy Samuels indicted?**

* + - Sili-Tech promised to deliver robots that would cause no bodily harm.
    - So Randy was not legally responsible for the death of the robot operator.
    - Waterson, the president of Sili-Tech has contributed large sums to the re-election of judge McMurdock, who indicted Randy.
    - Functional requirements specify the behavior of the robot under exceptional conditions. Operator intervention may be needed.
    - Exceptional conditions were not mentioned in training of operators.



**E-mail was not secure at Sili-Tech It was found that:**

* + - * Cindy Yardley faked the test results of the Killer code
      * Johnson told Cindy that the robot was safe and all will lose their jobs if the robot were not shipped on time
      * Randy stole some of his software, but admitted he had bugs in his Killer code
      * Email privacy
      * Professional ethics: delivering bug-free software
      * Plagiarism-intellectual property
      * Honor confidentiality
      * ACM code of ethics

Why is a software system any harder to create than a simple program?

* + - * Did George Cuzzins commit malpractice by choosing a lifecycle model that was inappropriate for the type of project?
      * What are the ethical implications of George Cuzzins firing Jan Anderson?
      * Not having the experience in any kind of interactive software development, should George Cuzzins have accepted the job?
      * Is the interface designer responsible for Bart Mathew’s death?
      * Since George Cuzzins chose a software lifecycle that did not emphasize the importance of the interface development, is he responsible for Bart Mathew’s death?
      * Was Max Worthington legal in revealing internal company documents?
      * Was Max Worthington ethical in revealing internal company documents?
      * Max Worthington was monitoring employee emails. Is this legal? Is this ethical?
      * Cindy Yardley faked results of tests, is she responsible for Bart Mathew’s Death?
      * What would you have done in Cindy’s place?
      * Ray Johnson “ordered” the tests to be faked, is he the MOST responsible for Bart Mathew’s Death?

##### Review of Facts



* + - * + FACT: Ray Johnson, Division Chief, threatened to layoff the entire robotics division
        + FACT: Ray Johnson announced the “Ivory Snow Theory”
        + FACT: Anonymous source, “Martha”, leaks inside information to paper
        + FACT: Randy Samuels did not accept criticism about his code.
        + FACT: Randy Samuels reportedly would storm out of meetings
        + FACT: George Cuzzins, Project Manager, decided to use the Waterfall Software Lifecycle against the advice of team members who recommended the Evolutionary Prototyping Software Lifecycle
        + FACT: George Cuzzins fires Jan Anderson after a disagreement about the project lifecycle model
        + FACT: George Cuzzins’ experience is in data processing projects
        + FACT: Dr. Gritty concludes that it was the interface design that killed Bart Mathews
        + FACT: Error message appears in green with an error number, no audio indicators - Inappropriate use of color

##### Review of Facts



* + - * + FACT: Emergency abort requires six separate sub menu choices

Poor visibility, poor hierarchy.

where is the big red button?

* + - * + FACT: No feedback from entry
        + FACT: Reference book is needed to determine the error and appropriate response
        + FACT: Prof. Silber announces that tests results in SiliTech’s documents were different from the test results obtained from the actual code
        + FACT: Prof. Silber states that Black Box, White Box, and Integration Testing may have been faked or that version control failed
        + FACT: Cindy Yardley, SiliTech Software Tester, admits to faking test results.
        + FACT: Cindy Yardley accuses Ray Johnson, Robotics Division Chief, of asking her to fake the tests.
        + FACT: Electronic Monitoring revealed that Ray Johnson was involved in faking tests and that George Cuzzins knew that the robot was unsafe

##### Review of Facts



* + - * + FACT: Electronic Monitoring revealed that Randy Samuels “lifted” code from another source and used it in the swing arm routine.
        + FACT: Max Worthington, Chief of Security, is “Martha”, and released internal email documents to the News Media.

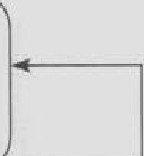
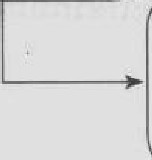
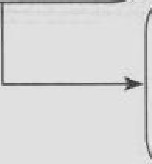
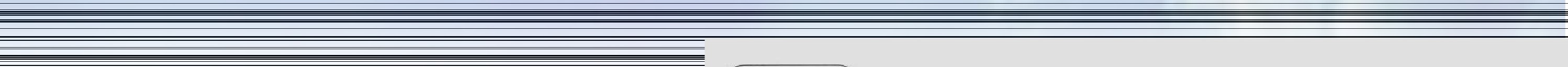


### Software Development Lifecycle

* Defines the basic, orderly steps necessary to guide a project from start to finish.
* Relates all schedules and work process to a specific sequence during the development project
  + Definition/Requirements Analysis
  + Conceptual/Interface Design
  + Program Design
  + Implementation/Coding
  + Unit Testing
  + Integration Testing
  + Acceptance Testing
  + Delivery
  + Maintenance



**Evolutionary Prototyping Model**



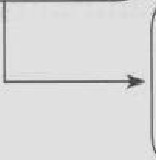
**Waterfall Lifecycle**

Software Concept

Requirements Analysb

Detailed Design

Architectural Design



Sy tt'm Testing

Coding :ind Debugging

nmplel •



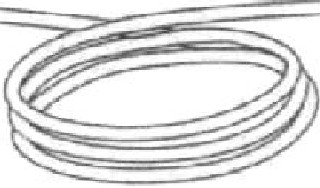
fi ne prococyp •

uncil accept. ble

|  |  |  |
| --- | --- | --- |
| Jnili I oncepr | l)esign and  11nrl menr | He |
|  | inirial |  |
|  | pmrucype |  |
|  |  |  |
|  | |

and rele:1:,,;

pmtul}"f)t:'





**Case of the Killer Robot**

**Article: Software Engineer Challenges Authenticity of “Killer Robot” Software Tests**

* **White Box Testing**
* Testing conducted with tests constructed from viewing the actual code.
* Each pathway through the code is tests, all end conditions are tested.

##### Black Box Testing

Testing conducted with tests constructed from the software specifications, but without knowledge of the code

##### Integration Testing

Testing conducted when separate code modules are integrated together into a single system

Tests constructed from the overall specifications of the system

##### Version Control

Methods for making sure the most current version of each module is included in the current version of the system build.

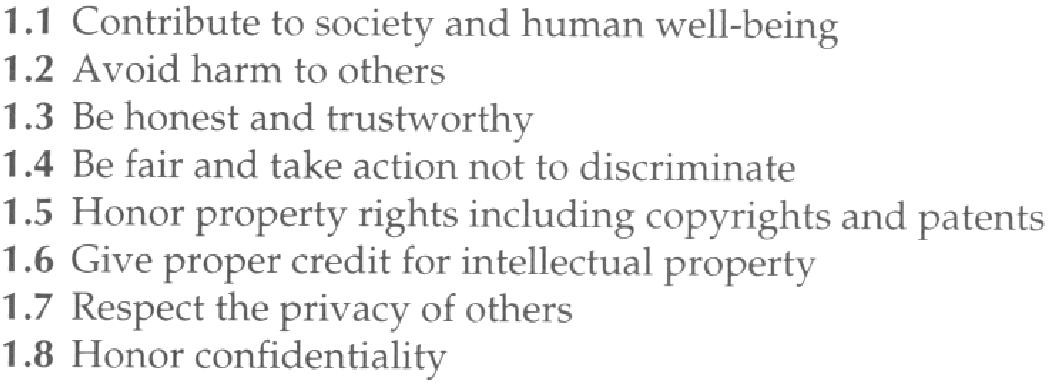
Allows development to “roll-back” to earlier versions



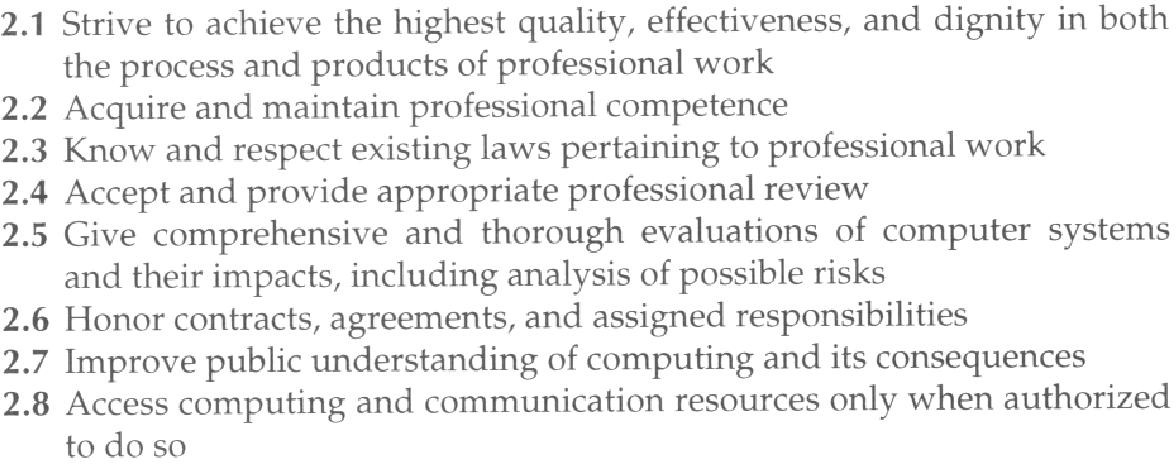
**Case of the Killer Robot**

**Article: Silicon Techtronics Employee Admits Faking Software Tests**

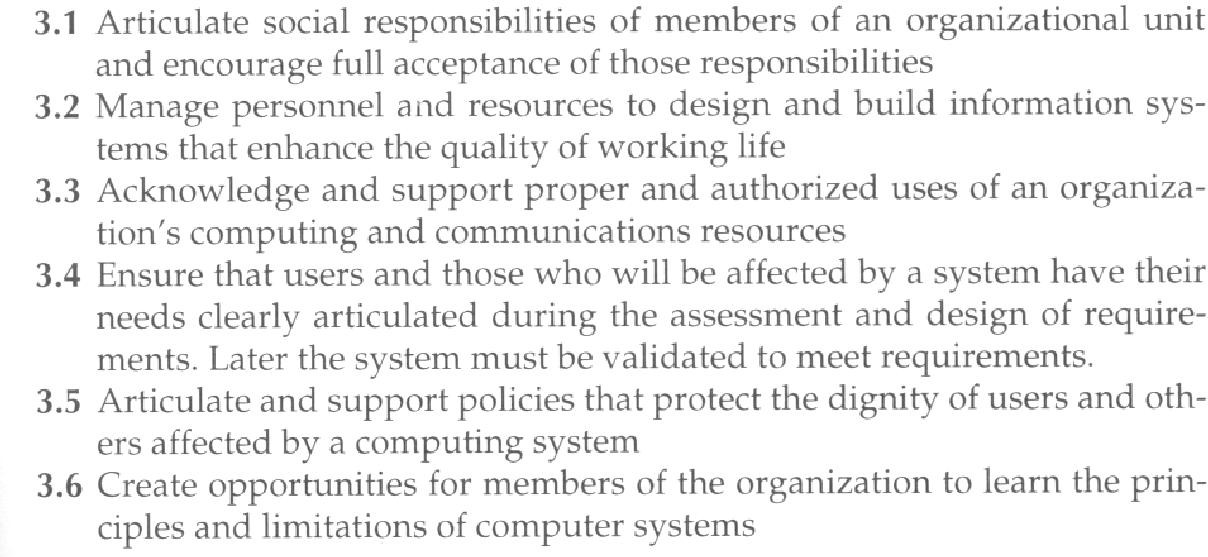
* + - * George Cuzzins knew the robot had problems but shipped it anyway, is he responsible for Bart Mathew’s Death?
      * Randy Samuels committed a criminal act of illegally using someone else’s code and knew there was a problem with the swing arm function, is he responsible for Bart Mathew’s Death?
      * Is Micheal Waterson, SiliTech CEO, responsible for Bart Mathew’s death?
      * Is Bart Mathew’s responsible for Bart Mathew’s death?
      * Dr. Yoder: “Randy Samuels and Cindy Yardley were both just out of school. … One has to wonder whether they received any instruction on ethics”
      * Before this class, what have you learned about ethics in the CS Department?
      * What responsibility does the CS Department have in teaching ethics?
* Dr. Yoder: “Related to this is the question of whether either of them had much prior experience with group work.”
* What have you learned about working in a group in the CS Department?
* What should the CS Department do to prepare you for working in a group?
* Dr. Yoder: “Samuels did not have requisite domain knowledge.”
* What is “domain knowledge”?
* Should having domain knowledge be a condition of employment for a programmer ?
* What is the ACM?
* What is the ACM Code of Ethics?
* General Moral Imperatives



* More Specific Professional Responsibilities



* Organizational Leadership Imperatives





**What ethical issues might you face?**

* Are you the kind of team member you would want on your own team?
  + Accept criticism (egoless programming)
  + Be aware of the difference between constructive and destructive criticism
  + Meet your team obligations
    - Be a contributing member
    - Meet deadlines
    - Provide help to others
  + Communicate
  + Be honest
* Are you the kind of team leader that you would want to lead your team?
  + How do you motivate your team to be productive?
  + Do you recognize your own lack of expertise?



**What ethical issues might you face?**

– What will be your leadership style?

* + - Autocratic
      * unlimited power is held by a single individual
    - Authoritarian
      * enforces strong and sometimes oppressive measures
    - Democratic
    - Laissez-Faire
      * The basic idea is that less government interference makes for a better system
* What ethical responsibilities do you have to the public (users of your programs)?
  + Thorough design, Robust implementation, Thorough risk analysis, Thorough testing, Proper training, Honesty in advertising