In 1206, the Arab engineer Al-Jazari invented a programmable drum machine where a musical mechanical automaton could be made to play different rhythms and drum patterns, via pegs and cams. Machine code was the language of early programs, written in the instruction set of the particular machine, often in binary notation. Computer programmers are those who write computer software. Later a control panel (plug board) added to his 1906 Type I Tabulator allowed it to be programmed for different jobs, and by the late 1940s, unit record equipment such as the IBM 602 and IBM 604, were programmed by control panels in a similar way, as were the first electronic computers. Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards. Use of a static code analysis tool can help detect some possible problems. Allen Downey, in his book How To Think Like A Computer Scientist, writes: Many computer languages provide a mechanism to call functions provided by shared libraries. Different programming languages support different styles of programming (called programming paradigms). Many applications use a mix of several languages in their construction and use. Implementation techniques include imperative languages (object-oriented or procedural), functional languages, and logic languages. Debugging is a very important task in the software development process since having defects in a program can have significant consequences for its users. Different programming languages support different styles of programming (called programming paradigms). Techniques like Code refactoring can enhance readability. These compiled languages allow the programmer to write programs in terms that are syntactically richer, and more capable of abstracting the code, making it easy to target varying machine instruction sets via compilation declarations and heuristics. In the 9th century, the Arab mathematician Al-Kindi described a cryptographic algorithm for deciphering encrypted code, in A Manuscript on Deciphering Cryptographic Messages. Compilers harnessed the power of computers to make programming easier by allowing programmers to specify calculations by entering a formula using infix notation. Use of a static code analysis tool can help detect some possible problems. Also, specific user environment and usage history can make it difficult to reproduce the problem. Many applications use a mix of several languages in their construction and use. Some languages are more prone to some kinds of faults because their specification does not require compilers to perform as much checking as other languages. Some languages are more prone to some kinds of faults because their specification does not require compilers to perform as much checking as other languages. Popular modeling techniques include Object-Oriented Analysis and Design (OOAD) and Model-Driven Architecture (MDA). Whatever the approach to development may be, the final program must satisfy some fundamental properties. However, readability is more than just programming style. One approach popular for requirements analysis is Use Case analysis.