Synopsis:

On May 11, 1996. ValuJet Flight 592 crashed into the Florida Everglades shortly after takeoff from Miami International Airport, killing all 110 passengers, crew members, and pilots onboard. Immediately following the incident, the NTSB took responsibility for finding the cause of the crash, and for finding ways of improving safety going forward. The cause of the crash was simple to identify and would be traced back to a fire that broke out in the forward storage compartment, where boxes containing hazardous oxygen generators were stored. These generators were typically housed above the passengers of a plane and would deliver oxygen in the event of a loss of pressure.

Following the incident, the NTSB tracked down the source of the problem to a contracted maintenance company, SabreTech. This company in turn contracted out mechanics to handle work that should have been done by internal employees. As a result, improperly handled and processed boxes filled with Oxygen generators ended up onboard Flight 592. While onboard, one (or more) of the canisters must have accidentally began leaking, and in the pure oxygen environment, a fire erupted. The cause of the fire was most likely due to the excessively high temperatures that the canisters reached, due to their improper storage conditions.

In the wake of the Tragedy, the NTSB investigated how this issue can be alleviated going forward from here. The problem was that this issue did not arise from a single incompetent move, but from a combination of several moves, that led to a catastrophic result. As for who *exactly* was at fault? It was difficult to tell. The agency managed to track down the root of the issue to several mechanics and managers who pencil-whipped the canisters without going through proper procedure. These canisters could have potentially been caught, had procedure been followed. The problem is made more complicated by the fact that the procedure was not written for a non-engineer audience, and the jargoned technobabble used in the manuals would have been incomprehensible to the technicians working for SabreTech.

Analysis:

The real question at hand for this disaster is clearly:
"Who was at fault, and what changes can be made to prevent this from happening again?" Unfortunately, there is no clear answer for either of these questions. Regarding the first question at hand, the disaster will never be pinned to just one person. It is a result of a systemic problem, a "system accident" as the author calls them. Not only are such problems difficult to discern blame for, but they're also very difficult to counteract and correct for. In this particular case, the blame could easily be set upon a small group of people, but the steps to prevent this disaster were not only unclear; but were also riddled with confusion regarding where the crucial failure took place. The author seems to believe that there were several main problems

that led to this failure:
1) SabreTech did not poses the necessary Safety Caps to be placed onto the firing pins, for transport of the Oxygen generators.
2) SabreTech's technicians did not properly inspect the generators, and pencil-whipped the signing-off for the devices
3) SabreTech's managers were unaware of the contents of the box containing the generators, and the hazardous devices were not stored properly.
4) The boxes were given to the shipping clerk without proper knowledge of the box's content.
5) The box containing the generators was mislabeled as "Airplane Parts"
6) Finally, the parts were not properly stowed within flight 592 due to the lack of safety notifications on the boxes/paperwork pertaining to the generators.

The GOOD thing about this list of complications is that several of them are directly caused by others. Issue #6 was directly caused by Issue #5, which in turn was the result of Issue #4, which in turn was the result of Issue #3, which was initially the result of Issue #2. One might believe that the blame can now be either cast on SabreTech for not providing safety caps, or on the technicians for not inspecting the generators; however, the issue runs deeper than that. Each issue following the inspection could have been alleviated, provided that the individual who caused it was aware of the danger that these generators presented. The Shipping clerk could have caught the issue before it progressed simply by doing research into what the unlabeled parts were in his office. The generic "airplane parts" could have been scrutinized by any number of technicians at the airport. The managers at SabreTech could checked off on the contents of the box containing the generators. However, I believe the real culprit is the disconnect between the engineers who wrote the technical manuals for these generators, and the poorly educated (and often non-English speaking) technicians who were expected to utilize these manuals. Technobabble is often times seen as a running joke in the engineering world, but more often than not, it can be harmful to not only the productivity of a company, but also to the safety and well-being of anyone interacting with a product. If the engineers who originally wrote the technical manuals for the Oxygen generators were to have a solid grasp on the mentality, mindset, thought process, and vernacular of the technicians who would be reading these manuals, the issue could have easily been avoided. The author does an exceptional job at painting the confusion that could have arisen (provided that the technicians attempt to read the manuals); saying that there would have been confusion between what canisters were "expired" versus "expended" and which were both, or neither. Even to experienced Engineers, the distinction between jargon is often times unclear. The use of formalities such as acronyms, trade terms, product names, and slang only intensifies this issue. Nobody inherently can discern the difference between a LCAC and a HVAC system just by looking at the terms. Just as no technician can discern the difference between an "Expired" versus an "Expended" Canister when pressed for time.

I believe that software is an even more ambiguous realm. Often times code is highly personalized, despite the best efforts of companies to generalize code. In my own experience, I often fell prey to the despicable “There is no documentation on that module” response when seeking guidance on a project. There are very few guidelines regarding exactly how code must be managed and organized within a company.

The fact that developing a project at any scale is such a fluid endeavor makes documentation all the more challenging. One can very easily say “This code snippet was taken from *itemSorter.py*, lines 100-150”, but this solution only works until *itemSorter.py* is replaced with *fileSorterExtraordinaire.py* at which point your documentation is meaningless. Although developers tend to fervently despise guidelines surrounding their heavily-personalized development process, it seems evident that the industry should move to implement some form of standardized documentation, in order to actively work to prevent tragic errors such as ValuJet Flight 592.

A prime example where code documentation most likely caused the failure of a whole system was with the Mars Climate Orbiter. The MCO was a NASA/Lockheed Martin joint satellite that was sent to Mars in order to study the Martian climate and atmosphere. In September of 1999, the craft unexpectantly and suddenly de-orbited due to a lack of communication between Lockheed and NASA. According to the agreement, Lockheed was expected to send their telemetry data to NASA’s ground station in Netwon-Seconds, however Lockheed’s own internal code for the orbiter was using Pound-Force-Seconds. This discrepancy of units led to the craft passing too close to Mars’ thick atmosphere, resulting in the craft disintegrating upon an unexpected entry. On the surface, one would expect the blame to fall on Lockheed for not providing their units as required by NASA. However, NASA places the blame on themselves, for not properly testing flight hardware prior to launch.

The lessons learned from tragedies such as the MCO, or ValuJet Flight 592 are unfortunately quite unclear. There is no simple takeaway that will allow all of us to be better engineers. My advice however, is to become passionate about your projects to the point where you are intrigued about every aspect of design. This familiarity with the entire design will allow you to catch issues that would otherwise slip under the radar (pun intended, of course).