ECSE Capstone Project Risk Analysis

Team #19

Semester 1, 2025

Identified Risks

Risk #1: Team Participation and Availability Issues

Risk Examples:

- Lack of communication and/or contribution.
- Team members are unable to attend key milestones or work sessions.
- Team member absences due to sickness or other schedule conflicts.

Impacts:

- Poor quality and last-minute work. Failure to contribute in severe cases, hence increasing workload for others.

Root causes:

- Unclear project expectations or imbalanced task distribution between members.
- Commitment to other courses and/or personal obligations outside of the university.
- Due to the size of the team, 1-2 members will likely be unavailable at any given time.

Risk Planning and Mitigation

1. **Set Clear Expectations**: Define participation and task responsibilities and encourage proactive communication within the team about absences and workload.
2. **Use Tools for Tracking and Accountability**: Use GitHub, Google Docs and a progress tracker, e.g. Trello, to track contributions and allow for handovers.
3. **Work Distribution and Support**: Assign small groups by specialisation, have specialisation leads check-in and reassign tasks as needed based on team engagement and absences.
4. **Transparent and Accessible Communications**: Keep detailed meeting minutes and ensure specialisation leads or project leads update absent members.

   Example: In ENGGEN 303, a member did not make an effort to join meetings for unexplained reasons, leading to a lack of understanding of deliverables. The team leader engaged with the member and communicated expectations. He offered support and used another platform with edit history resulting in the student beginning to contribute.

Monitoring and Escalation

1. **Flagging Low Engagement**: Missing two meetings, unexplained, will be flagged. Specialisation leads will track work output whilst considering workload balance with other courses.
2. **Direct Intervention and Support**: Specialisation leads to check-in with low-contributing members, offering guidance and support. Also includes adjusting the workload if necessary.
3. **Escalation to Mentor**: If participation issues continue to occur despite check-ins, the project lead discusses them with the team before escalating to the mentor.

Risk #2: Equipment, Component or Microcontroller Damage

Risk Examples:

- Electrostatic discharge (ESD) causes damage to the microcontroller or components.
- Incorrect wiring, e.g., short circuits or polarity reversals, can damage the device.
- Computer damaged from accidental mishandling.

Impacts:

- Financial and time costs needed to source and resolder new components.
- Loss of unsaved work leads to wasted time and effort.

Root causes:

- Exposure to extreme voltage, temperature, ESD, or incorrect wiring.
- No tolerance for user error, e.g. installing components the wrong way around.
- Accidental damage due to mishandling, e.g. water/drop damage and theft.

Risk Planning and Mitigation

1. **Secure Equipment Storage**: Store equipment safely in lockers after use.
2. **Component Availability**: Keep track of ECSE store hours to accommodate access to new/replacement components.
3. **Implement Best Practices**: Follow proper electrical equipment handling, e.g. anti-static wristbands.
4. **Design for Fault Tolerance**: Design PCB with protection features for incorrect use, e.g. zener diodes for overvoltage protection and polarity inversion protection.
5. **Incremental Design Approach**: Solder PCB in stages and verify that previous stages function before soldering.
6. **Backup Testing Plan**: Have a backup testing plan using software simulations, e.g. Postman for API testing.
7. **Frequent Backups**: Commit and push changes to a remote git repository regularly.

   Example: In EE209, many teams damaged components by neglecting the power supply's polarity and voltage. We mitigated this by adding ‘surge protection’ sacrificial zener diodes and by probing the circuit during soldering to find mistakes as they happen. Another team stored their project files on a USB stick, which failed, causing them to fail to deliver their project. Our team committed to GitHub regularly to avoid data loss and to facilitate transparency of work.

Monitoring and Escalation

1. **Component Management**: Track component usage and visit ECSE stores when necessary to request replacement parts.
2. **Immediate Reporting**: If the microcontroller is damaged, inform the teaching team immediately.

Risk #3: Misuse of Generative AI Tools

Risk Examples:

- Reliance on AI for report writing or coding leads to academic integrity violations.

Impacts:

- Members create poor-quality work that may be flagged by AI detectors.

Root causes:

- Students may rely too much on AI tools instead of comprehending the material.

Risk Planning and Mitigation

1. **Encourage Communication**: Promote team members to seek help from the group.
2. **Peer Review**: Review submissions to ensure plagiarism-free submitted work.

   Example: In CS301, a team member used AI to generate the code. Unfortunately, the AI produced incorrect code, and the member did not understand the code. We sat with him and talked through expectations. He then transferred the task to another team member who wrote the algorithm instead.

Monitoring and Escalation

1. **Issue Escalation**: If a student continues misusing AI and compromises the project, the issue is escalated to the teaching team.

Risk #4: Integrating Three Different Disciplines within the Team

Risk Examples:

- Invalid database values and foreign keys interrupt firmware-software integration.
- Data constraints are not enforced, leading to invalid data handling.
- Raspberry Pi sends malformed data to the wrong endpoint.

Impacts:

- We may have a non-functional product due to components not working together.
- Major restructuring/rewriting may be required to ensure parts compatibility.
- Potential disagreements within the team on the best approach towards the project.

Root causes:

- Miscommunications between different specialisation teams (specialisation teams).
- System components are not modular.
- Lack of inter-team communication at the design stage and throughout the project.

Risk Planning and Mitigation

1. **Database Constraints**: Establish constraints to enforce data validity.
2. **Inter-Team Discussion**: Ensure active discussion between all specialisation teams.
3. **Error Logging and Debugging**: Implement logging to detect malformed data.
4. **Technical Research**: Read Raspberry Pi and other relevant documentation.

   Example: The software team can implement an API expecting data in a JSON format, but the firmware team has opted for a different format, e.g. CSV, causing integration failures. This can be mitigated by maintaining open communication, shared documentation and regular check-ins.

Monitoring and Escalation

1. **Regular Log Reviews**: Review logs frequently to address issues before they escalate.
2. **Decision Tracking**: Log significant decisions made in the meeting minutes.
3. **Proactive Integration Discussions**: In meetings, talk specifically about what people are working on and plans for integration to identify potential integration issues.

Risk #5: Safety in the E&I Labs

Risk Examples:

- Electrical hazards, e.g. not checking the soldering for short circuits.
- Improper soldering, e.g. extreme temperatures, unclean tools, no soldering flux.
- Chemical exposure, e.g. inhaling flux while soldering.

Impacts:

- Team members may suffer injuries or develop health issues.
- Lab access is restricted till members meet health and safety guidelines.

Root causes:

- Lack of proper safety training.
- Ignoring standard lab safety procedures.
- Not reading component datasheets.

Risk Planning and Mitigation

1. **Work in Pairs**: Always work with partners in hazardous areas (i.e. labs).
2. **Equip Proper PPE**: Wear proper PPE, e.g. closed-toe shoes and safety glasses.
3. **Ventilation**: Use a fume extractor and safety glasses when soldering.
4. **Safety Training**: Ensure team members undergo safety training for lab safety.

   Example: From EE209 and EE311, many teams had soldered the components at a very high temperature (exceeding the maximum temperature of the component) or forgot to turn the fume extractor on before soldering. This can be mitigated by having a pair of members go to the soldering room and ensuring that safety procedures are followed before soldering.

Monitoring and Escalation

1. **Enforce Safety Measures**: Ensure members follow lab safety policies and complete induction training before prototyping.
2. **Training Verification**: Confirm members responsible for soldering have read datasheets and completed the necessary safety training.
3. **Immediate Reporting**: If any safety violation occurs, report the incident.

Risk #6: Poor Product Performance and Design Choices

Risk Examples:

- Failure in data transmission due to poor range and unexpected disconnection.
- Design malfunction and manufacturing delay due to substandard PCB.

Impacts:

- Rushed work leads to poor design or, in severe cases, failure to submit the design.

Root causes:

- Hardware component tolerances and ambient effects., e.g. resistor values tolerances, temperature effects on components, ambient WIFI/Bluetooth noise.
- Lack of hands-on experience with PCB design and PCB design software.

Risk Planning and Mitigation

1. **Robust Design and Testing**: Simulate designs using software tools before manufacturing, prototype using breadboards and get TA approval for PCB before manufacturing.
2. **Data Reliability Measures**: Use data error detection techniques, e.g., checksums.

   Example: In CS301, bluetooth issues due to noisy power supply were solved early during breadboarding. Decoupling capacitors were used for every IC.

Monitoring and Escalation

1. **Address Recurring Failures**: Explore alternative protocols, e.g. Bluetooth.
2. **Checklist**: Create a design checklist and escalate major flaws to mentors and TAs.

Risk #7: Unavailability or Delays in Sourcing Components/Software

Risk Examples:

- Supplier issues, outdated component lists, or software libraries become deprecated.
- Miscommunication about design specifications or unexpected lead times.

Impacts:

- Delayed delivery of components affecting project completion.
- Unable to experiment with unique components or software to learn how it works.

Root causes:

- Manufacturing bottlenecks or supply chain delays.
- Errors in design files require corrections.

Risk Planning and Mitigation

1. **Early Acquisition**: Get PCB components in advance and confirm supplier timelines.
2. **Design Verification**: Double-check design files before submission to avoid errors.
3. **Flexible Component Selection**: Use through-hole components for more adaptability to allow other packages.

   Example: In EE209, we picked an unavailable component (through-hole TRIAC). We discovered the issue by trying to source the component for our breadboard. We had to change our design to use a different component.

Monitoring and Escalation

1. **Supply Tracking**: Track supplier progress and follow up on delays.
2. **Escalation Plan**: If the delay is longer than expected, escalate to the project supervisor for guidance or alternative solutions.

Risk #8: Scope Creep

Risk Examples:

- Team members are stuck implementing features that may not be required.
- Considering new features instead of refining or fixing bugs.

Impacts:

- The product delivered has many features, but none of them work.

Root causes:

- Client's demands may be unclear or excessive, e.g. monitoring temperature may be difficult to do accurately.

Risk Planning and Mitigation

1. **Minimum Viable Product (MVP)**: Create a list of features needed for an MVP.
2. **Feature Evaluation**: Discuss with the specialisation lead and team before implementing new features.

   Example: We made an extra website for EE209, which wasted 2 weeks of one member's time.

Monitoring and Escalation

1. **Focus on other tasks**: Reallocate people to focus on other deliverables and documentation instead of implementing new features.