# Capstone Module_1 - Defect Management of an Android Watch

---

Group_2:

1 -> Hrushikesh
2 -> Harshvardhan
3 -> Asha
4 -> Shilpa
5 -> Vinith
6 -> Jayavardhini

---

## Defect Identification:

- In addition to the typical sources, pay close attention to defects related to battery life, connectivity (Bluetooth, Wi-Fi), and user interactions specific to the watch form factor (e.g., touch gestures, voice commands).
- Categorize defects based on their impact on the watch's core features, such as fitness tracking, notifications, and app functionality.

## Defect Analysis:

- Analyze defects in the context of the watch's hardware limitations, such as processing power, storage capacity, and display size.
- Consider the impact of defects on the watch's integration with other devices or ecosystems (e.g., smartphones, fitness apps).

## Defect Assignment and Tracking:

- Involve cross-functional teams, including hardware engineers, firmware developers, and user experience designers, as defects may span multiple domains.
- Prioritize defects based on their impact on critical watch functions, such as timekeeping, health and fitness tracking, and user safety.

## Defect Resolution:

- Coordinate with hardware and firmware teams to address defects that may require hardware or low-level software changes.
- Ensure that fixes account for the watch's resource constraints and minimize performance impacts.
- Test fixes thoroughly on different watch models and configurations, as well as in various usage scenarios (e.g., during workouts, while receiving notifications).

## Defect Verification and Closure:

- Conduct verification testing on actual watch devices, as emulators or simulators may not accurately represent real-world usage conditions.
- Involve end-users or beta testers to validate fixes and gather feedback on the overall user experience.

## Continuous Improvement:

- Analyze defect data to identify patterns or trends related to specific watch models, hardware revisions, or software versions.
- Collaborate with hardware and firmware teams to improve the overall quality and reliability of the watch platform.
- Implement processes for over-the-air (OTA) updates to address critical defects or security vulnerabilities promptly.

## Communication and Collaboration:

- Ensure clear communication and alignment between the watch development team and teams responsible for companion apps or services (e.g., smartphone apps, cloud services).
- Collaborate with external partners or third-party developers to address defects in their watch applications or integrations.

## Resolving Defects:

- Assign defects to developers for investigation and resolution.
- Implement a fix and update the bug report with the solution.
- Thoroughly test the fix on actual watches and emulators to avoid regressions.

## Verification and Release:

- After a fix is implemented, create a new software update and test it rigorously internally on watches.
- Consider a limited beta release to a small group of users for further verification on real-world usage.
- Once confirmed stable, release the update to the general public through the official app store.