Republic of the Philippines
Sorsogon State University
COLLEGE OF INFORMATION AND COMMUNICATIONS TECHNOLOGY
Bulan Campus
Zone-8, Bulan, Sorsogon
Tel. No.; (056) 311 9800; Email Address: cict@sorsu.edu.ph

TRICYCLES: ENDING UNFAIR PRICING WITH LOCATION-BASED RATES IN BULAN, SORSOGON

TECHKEYS

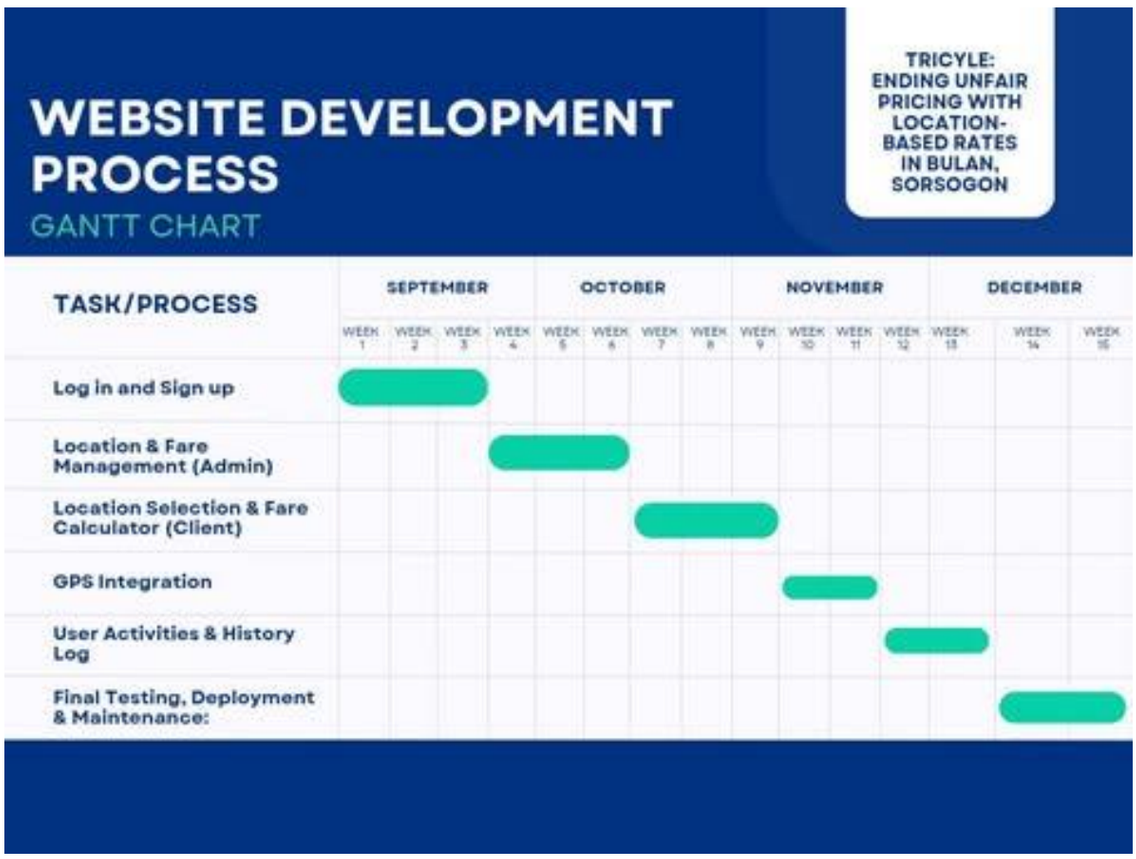
VIVIAN JOY ELEMENTO

SEAN PAUL SANTOS

EUGENE ESMERIA

BSCS 4-2

Republic of the Philippines
Sorsogon State University
COLLEGE OF INFORMATION AND COMMUNICATIONS TECHNOLOGY
Bulan Campus
Zone-8, Bulan, Sorsogon
Tel. No.; (056) 311 9800; Email Address: cict@sorsu.edu.ph



This **Gantt Chart** presents the website development process for the tricycle fare management system in Bulan, Sorsogon, scheduled over 15 weeks from September to December. The project begins with the creation of the log in and sign-up feature, followed by the development of the location of the fare management system for the admin. Next, the client-side location selection and fare calculator are built, leading to the integration of GPS for accurate tracking. Afterward, user activities and history log are implemented to monitor interactions. Finally, the process concludes with final testing, deployment, and maintenance to ensure the system is fully functional and reliable.

“SDLC Model (SCRUM)”

Sprint 1: September (Week 1–3)

- **SDLC Phase:** Planning + Requirement Analysis
- **Tasks:**
  - Define requirements & user stories (Login, Fare Mgmt, GPS, History).
  - Setup tools (Laravel, Vue.js, Git, CI/CD pipeline).
- **Deliverable:** Product backlog, sprint backlog, initial UI wireframes.

Sprint 2: September (Week 2–4)

- **SDLC Phase:** Design + Development (Iteration 1)
- **Tasks:**
    - Develop **Log in & Sign up** module.
    - Database schema for users.
    - UI/UX login/signup pages.
- **Deliverable:** MVP authentication module.

Sprint 3: October (Week 5–6)

- **SDLC Phase:** Development (Iteration 2)
- **Tasks:**
    - Build **Location & Fare Management (Admin)**.
    - Set fare tables & routes.
    - Admin dashboard mockups.
- **Deliverable:** Admin panel with fare setup.

Sprint 4: October (Week 7–8)

- **SDLC Phase:** Development (Iteration 3)
- **Tasks:**
    - Implement **Location Selection & Fare Calculator (Client)**.
    - Pick-up/drop-off selection with fare computation.
    - API for fare calculation.
- **Deliverable:** Client fare calculation module.

Sprint 5: November (Week 9–10)

- **SDLC Phase:** Development (Iteration 4)
- **Tasks:**
    - Integrate **GPS tracking**.
    - Real-time location detection (e.g., Google Maps API).
- **Deliverable:** GPS integration.

Sprint 6: November (Week 11–12)

- **SDLC Phase:** Development (Iteration 5)
- **Tasks:**
    - Create **User Activities & History Log**.
    - Store trip history (location, fare, date/time).
- **Deliverable:** History log system.

Sprint 7: December (Week 13–14)

- **SDLC Phase:** Testing + Refinement

- **Tasks:**
    - Unit + integration testing.
    - Fix bugs from sprint reviews.
- **Deliverable:** Tested modules, QA report.

Sprint 8: December (Week 15–16)

- **SDLC Phase:** Final Testing, Deployment & Maintenance
- **Tasks:**
    - Final system testing & UAT.
    - Deployment to live server.
    - Maintenance plan setup.
- **Deliverable:** Deployed system + documentation.

Justification:

We chose **Agile** for this project because it allows us to work in smaller parts called sprints where we build and test features step by step. This is important since our system has different users like drivers, passengers, and admins who may give feedback as we go along. Instead of waiting until the very end to see if everything works, Agile makes it possible to deliver working features early such as the login system first, then the fare calculator, and then the GPS. It is also flexible because if the fare rates or rules in Bulan change, we can adjust the system in the next sprint without starting over. This way, the project becomes more focused on the users, less risky, and ensures that what we build really matches the needs of the community.

We are using **Scrum** because it allows us to build the system step by step through sprints, making it easier to test and improve along the way. It also helps us adjust to changes and get feedback from users like drivers and passengers, so the final system is more useful and practical**.**