BULAN PROFILING RESIDENT AND POPULATION MANAGEMENT SYSTEM
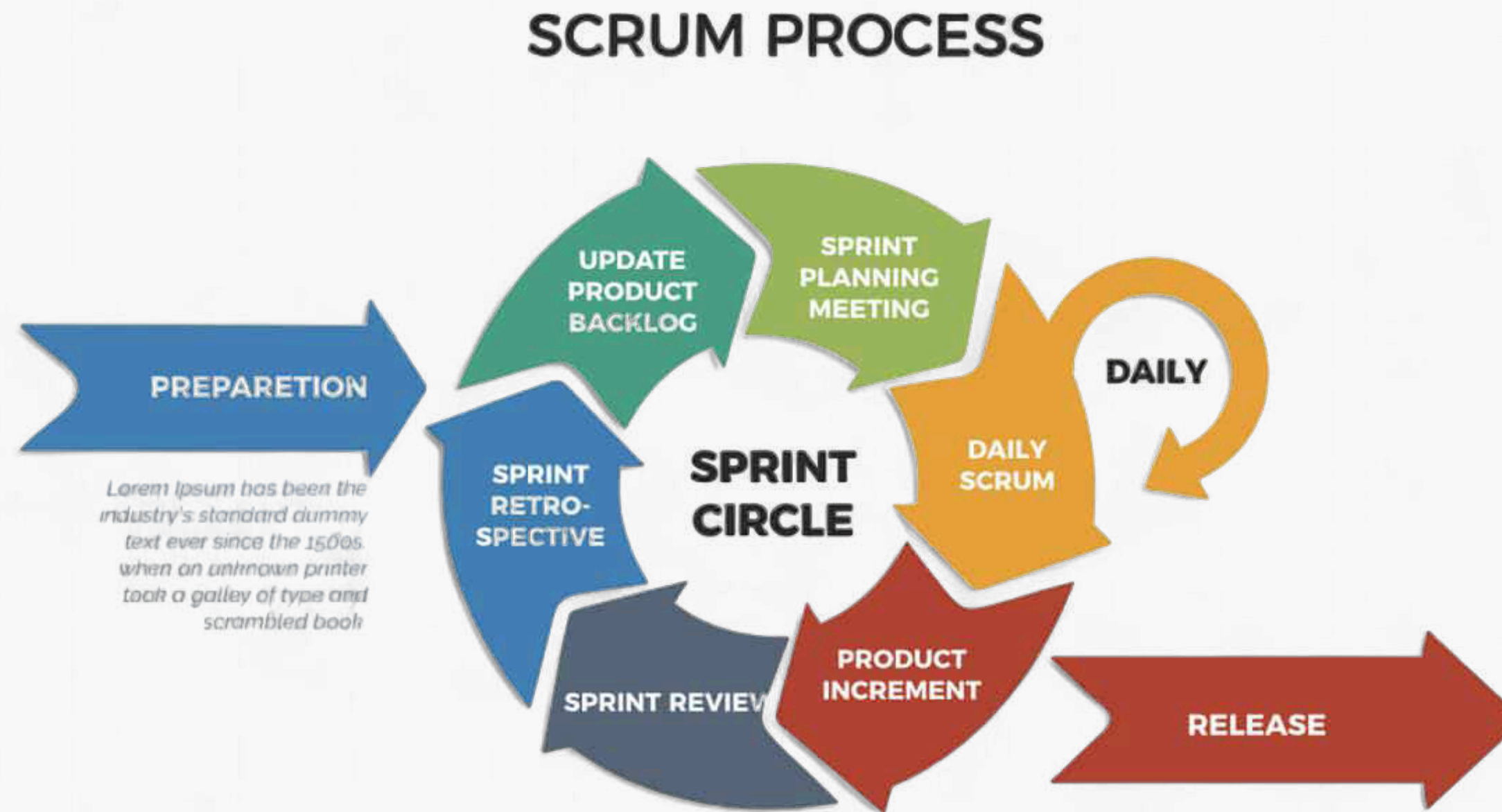
(SDLC AND GANTT CHART)

ADOPTED SDLC MODEL:
AGILE (SCRUM FRAMEWORK)

The Agile Scrum methodology is the most suitable SDLC model for the bulan Resident Profiling and Population Management System because:

- **Changing Requirements-** Barangay population data changes frequently (new residents, migration, births, deaths). Scrum allows flexibility to adjust requirements each sprint.

- **Continuous Feedback -** Barangay officials and health workers can provide feedback every sprint to refine the system according to their needs.

- **Incremental Delivery -** Features like resident registration, health records, and reporting can be delivered in increments, so users benefit early without for the whole system.

- **Collaboration** - Promotes teamwork between developers, barangay officials, and stakeholders, ensuring the system addresses real community nee

Risk Reduction - Regular sprint reviews and retrospectives help identify and fix problems early reducing the risk of failure

SOFTWARE DEVELOPMENT LIFE CYCLE (SDLC) AGILE

SCRUM PROCESS APPLIED TO THE SYSTEM

Sprint Planning Meeting

- Define sprint goals (e.g., "Build resident registration module").
- Create product backlog tasks: forms, database schema, search fuction ets.

Product Increment

- Deliver working features such as:
  - Adding resident profiles
  - Generating barangay population reports

  Updating household information

Sprint Retrospective

- Reflect on what worked and what needs improvement in the process.

Release

- Deploy the stable version to the barangay for real use after successful testing.

Preparation

- Gather requirements from LGU, barangay captains, and health workers.
- Identify core features: resident profiling, population statistics, household records, and report generation.

Daily Scrum

- Team members discuss progress, blockers, and next steps.

Sprint Review

- Demonstrate the new feature to barangay officials and health workers workers for feeback

Update Product Backlog

- Adjust tasks for the next sprint based feedback

We chose Agile-scrum because it:

- Provides **flexibility** for evolving requirements.
- Ensures **continuous improvement** through feedback loops.
- Delivers **early results** (barangays can test modules immediately).
- Reduces r**isks** by catching errors early in each sprint.

#	Tasks Name	Timeline	Description
1	Product Backlog Creation	Sep 4	Gather requirements, identify resident data needs, set up backlog
2	Sprint 1: Planning	Sep 11	Prioritize task(resident registration & login)
3	Sprint 1: Sprint Backlog	Sep 11	Select sprint tasks (resident form, login system
4	Sprint 1: development & Daily Scrum	Sep 18	Build resident profile + Login, daily standups
5	Sprint 1: Sprint Review	Sep 25	Demo login + profile form to LGU
6	Sprint 1: Sprint Restrospective	Sep 25	Team refection & backlog adjustment
7	Sprint 2: Planning	Oct 1	Set goals for CRUD + search features
8	Sprint 2 : Sprint Backlog	Oct 1	Select data management tasks
9	Sprint 2: Development & Daily Scrum	Oct 1	Build CRUD operations + search/ filter
10	Sprint 2: Sprint Review	Oct 8	Demo CRUD functions
11	Sprint 2: Sprint Retrospective	Oct 8	Reflect on Sprint 2
12	Sprint 3: Reports & analytics	Oct 15	Create population statistics & export features
13	Sprint 4: Security & Roles	Oct 29	Add roles (Admin, Barangay, Viewer),implement privacy compliance
14	Sprint 5: Optimization & Alpha Test	Nov 8	Internal testing, bug fixing, performance tuning
15	Sprint 6: Beta Test & Adjustments	Nov 22	Beta test with barangay officials, integrate feedback
16	Release & Training	Dec 5	Deploy final system, Train LGU & barangay staff, documentation
17	Buffer/ Wrap-up	Dec 19	Risk buffer & year end monitoring