**Implementation - Developing the Agile Plan**

**Phase 1: Planning and Requirements Gathering**

Duration: 2 Weeks

**Objective:** Define the vision, goals and features of the CAD Pay app; prioritize MVP features and align stakeholders.

**Activities:**

* Kick-off Meeting: Align the stakeholders on goals, timelines, and expectations.
* Requirement Gathering:
* Interview stakeholders to elicit all functional and non-functional requirements.
* Develop user personas with their pain points. Examples include frequent money transfers and currency support.

**Epic and User Story Creation:**

* Break the requirement down into epics and user stories.
* Prioritize user stories in the backlog using MoSCoW (Must, Should, Could, Won't).
* Initial Architecture Design: Design the architecture of the system for backend, frontend, and APIs.

**Deliverables:**

* Product vision document.
* Prioritized backlog with epics and user stories.
* Initial system architecture diagrams.

**Phase 2: Design and Prototyping**

Duration: 4 weeks

Objective: To deliver UI/UX designs intuitive to understand, embrace, and deliver technical design documents.

**Activities:**

* User Flow Design.
* Steps that map user journeys relevant to all major features, that include money transfer, payment setup, admin panel.

**Wireframes and Mockups:**

* Wireframe for the main screens of the app, that is Home, Send Money, Admin Dashboard.
* Create the high-fidelity prototypes for user testing.

**Feedback Loops:**

* Design Review to be done with the stakeholders.
* Feedback from stakeholders and potential users to iterate on.

**Technical Design:**

* API contracts for the integration documentation such as payment gateways, currency exchange API
* Schema diagrams for the database

**Deliverables:**

* Final UI/UX designs and style guide.
* Database schema and API documentation.

**Phase 3: Development (MVP Build)**

Duration: 8 Sprints (16 weeks)

**Objective**:

* Build core features incrementally.
* Sprint Plan
* Sprint Goals Deliverables
* Sprint 1 Set up a development environment and CI/CD pipelines for a ready-to-use development environment and pipelines.
* Sprint 2 Develop user authentication- login, signup, password recovery: Authentication feature.
* Sprint 3 Develop the Send Money feature backend and basic UI: Basic money transfer functionality.
* Sprint 4 Implementation of payment gateway integration- Stripe/PayPal: Payment gateway integrated.
* Sprint 5 Develop Transaction History Module: Transaction history page up and functional.
* Sprint 6 Develop Payment Setup Module: Payment Method Management Feature.
* Sprint 7 Admin Panel implementation for transaction management: Admin Dashboard.
* Sprint Activity to be performed Outcome
* 8 Add currency conversion feature using APIs: Fully-fledged multi-currency support.

**Activities per Sprint**-

* Sprint Planning: Define the sprint's goal and allocate assignments.
* Daily Stand-Ups: Quick meetings are where team members tell about the accomplished work and blockers.

**Development:**

* Front-end: Responsive User Interface using React/Flutter.
* Back end: Node.js for API based functionalities.

**Testing:**

* Unit testing performed by developers.
* Functional and integration testing performed by QA.

**Sprint Review:** Showing the work that has been accomplished to stakeholders.

**Sprint Retrospective:** To revisit successful events and discuss points of improvement.

**Deliverables:**

* Incremental builds of core functionalities in the application.
* Unit and integration tests for each module.

**Phase 4: Testing and Quality Assurance**

Duration: 2 Sprints (4 Weeks)

**Objective:** Ensure the app is stable, secure, and ready for release.

**Activities:**

* Functional Testing: Verify all user stories and acceptance criteria are met.
* Regression Testing: Test previously developed features after new ones are added.
* Performance Testing: Ensure the app performs well under load.
* Security Testing:
* Validate PCI-DSS compliance.
* Test for vulnerabilities, such as SQL injection and CSRF.
* User Acceptance Testing: Have testing sessions with stakeholders in order to get the last remarks.

**Deliverables:**

* Test cases and results.
* Bug reports and fixes.
* Approval for deployment from stakeholders.

**Phase 5: Deployment and Launch**

Duration: 2 Weeks

**Objective:** Deploy the app into production and ensure a smooth launch.

**Activities:**

* Staging Deployment: The deployment of the app happens in a staging environment for final checks.
* Final Testing: Smoke testing in a staging environment.
* Production Deployment: Deploy the application to production servers.

**Launch Plan Execution:**

* Publish mobile applications on App Store and Google Play.
* Announcement of launch through marketing channels.
* Onboarding Support: Provide tutorials and customer support for early adopters.

**Deliverables:**

* The live app will be accessible to the users.
* Documentation for onboarding users.

**Phase 6: Post-Launch Support and Iterations**

Duration: Ongoing

**Objective:** Improve the app based on user feedback and scale as necessary.

**Activities:**

Performance Monitoring:

* Track key metrics related to the app (for example, user retention and transaction success rate).
* Utilize analytics tools - Google Analytics, Firebase.
* Bug Fixes: Provide solutions to the bugs reported.
* Features Enhancements: Implement following features:
* Reward system,
* Go-Bus ticket booking,
* Advanced fraud detection,
* Scalable Infrastructure - servers and databases to deploy load,
* Continuous Delivery: Release: Build and deploy regularly.

**Deliverables:**

* Versions with an enhanced feature set of application.
* Reports of user testing and bug fixes.
* Agile Tools and Techniques

**Tools:**

* For task management: Jira, Trello.
* Version Control: Github/Gitlab.
* For design: Figma and Adobe XD.
* Testing - Selenium, Postman, Jest.
* Jenkins, CircleCI.

**Ceremonies:** Sprint Planning, Daily Stand-ups, Sprint Reviews, Retrospectives.

**Artifacts:** Product Backlog, Sprint Backlog, Increment.

This plan ensures the development of CAD Pay in a user-centric, iterative, and scalable way. It emphasizes regular feedback, continuous delivery, and adaptability to changing requirements.