Project Documentation

1. Project Title

Odoo E-Learning on Ubuntu Touch

2. Objective

The main objective of this project was to create a mobile application for Ubuntu Touch that seamlessly integrates with the Odoo OCA E-Learning module. The app allows users to perform key e-learning functions such as browsing and enrolling in courses, completing lessons and quizzes, and tracking their progress and certificates. This app addresses a gap in the Ubuntu Touch ecosystem, which currently lacks a dedicated Odoo E-Learning app.

3. Core Features

The application provides the following core functionalities:

- User Authentication: Users can log in using their Odoo credentials.
- Course Catalog: The app displays a list of available courses.
- Enrollment & Progress: Users can enroll in courses and track their completion status.
- Content Viewing: Users can view lessons, watch videos, and take quizzes directly within the app.
- Offline Mode: The app is designed to cache content, such as lessons and quizzes, for offline access using SQLite.
- Certificate Download: After completing a quiz, a user can generate and download a certificate.

4. Technical Stack

The app was built using the following technologies:

- **Frontend:** The user interface (UI) is built using **QML** and **JavaScript**, which is the standard for Ubuntu Touch UI development. The project includes various QML components for pages like the welcome screen, login, catalog, and quiz interface.
- **Backend:** The backend logic is handled by **Python** for interacting with the Odoo API. The current implementation includes a minimal, optional backend using Flask for demonstration purposes.
- **API:** The app uses the Odoo XML-RPC or REST API for fetching course lists, submitting quiz answers, and syncing progress.
- **Packaging:** The final deliverable is a `Clickable` app, which is the native packaging format for Ubuntu Touch.

5. Installation and Setup

To run this project, follow these steps:

1. **Set up Odoo:** Install a test instance of Odoo and the OCA E-Learning module.
2. **Set up Ubuntu Touch SDK:** Configure your development environment with the Ubuntu Touch SDK.
3. **Copy Files:** Copy the provided QML and Python source files into the project structure.
4. **Install Dependencies:** If you are running the optional backend, install the required dependencies using `pip install -r backend/requirements.txt` (Flask).
5. **Run:** The project is a runnable skeleton that can be launched with the Ubuntu Touch SDK.

6. Project Structure

The project is organized into logical components to manage the user flow from start to finish:

- `main.qml`: The main application window that manages stack-based navigation.
- `components/`: Contains individual UI pages like `WelcomePage.qml`, `LoginPage.qml`, `CatalogPage.qml`, and `QuizPage.qml`.
- `dialogs/`: Includes reusable dialogs for tasks like playing videos, showing certificates, and managing downloads.
- `utils/`: Contains utility scripts for API stubs, certificate management, and settings management.
- `backend/` (Optional): A minimal Flask backend for API simulation.
- `assets/`: Stores project resources like images and the `qml.qrc` file.

7. Limitations

- The `OdooAPI.js` and `SettingsManager.js` files currently contain stub functions. They need to be replaced with real network and storage code to fully integrate with a live Odoo instance.
- The authentication is a local simulation and requires real Odoo credentials for proper functionality.
- The offline mode uses a simple SQLite cache, but a more robust synchronization mechanism would be needed for production use.
- The certificate generation is a placeholder and would need to be expanded to produce real, verifiable PDFs.

8. Submission Details

- **Code Repository:** https://github.com/Tejas-952007/ubuntu-touch-phone-opreating-system-learning-app