

Telegram Bot Development: From Beginner to Pro

Telegram bot development from beginner to pro. Architecture, APIs, databases, and payments.

DevOps Instructor: Derek Holtzman • 5 sections • 49 lessons • 17.75h total

[View this course online](#) [Browse all courses](#)

About this course

This course takes you from beginner to proficient Telegram bot developer. You will learn bot architecture, API integration, database management, payment processing, and advanced bot features across multiple programming languages.

Course curriculum

Foundations and Preparation

- **How Telegram Bots Work: Core Concepts (12m)**
Understand the bot architecture, the BotFather, and the API. Learn the difference between a bot and a user account.
- **Setting Up Your Development Environment (8m)**
A step-by-step guide to installing Python, a code editor, and the essential libraries for Telegram bot development.
- **Creating Your First Bot with BotFather (15m)**
Practical walkthrough of registering a new bot, getting your API token, and initial bot configuration.
- **Choosing a Bot Framework: python-telegram-bot vs. aiogram (18m)**
Compare the two most popular Python libraries to decide which fits your project needs and coding style.
- **Understanding the Update Loop and Polling (10m)**
Learn how your bot receives messages and the basics of handling updates with polling methods.
- **Writing Your First Bot Script: Hello World (14m)**
Hands-on coding session to build a simple bot that responds to the /start command and sends a welcome message.
- **Basic Error Handling and Logging (9m)**
Learn how to catch common errors and set up logging to debug your bot effectively from the start.
- **Deploying Your Bot: Local vs. Cloud Hosting (20m)**
Explore options from running your bot on your own computer to using cloud services like Heroku or PythonAnywhere.

Core Bot Features and Commands

- **Handling Commands: /start, /help, and Custom Commands (16m)**
Implement a command handler structure to manage user interactions systematically.
- **Creating Interactive Menus with ReplyKeyboards (22m)**
Design user-friendly menus to guide conversations and collect input without free text.
- **Building Inline Keyboards for Actions and Navigation (25m)**
Learn to create buttons that trigger actions within messages, perfect for polls and quick replies.
- **Processing User Text Input and Callback Queries (19m)**
A problem-solving session on capturing user input and handling button clicks securely.
- **Sending Media: Photos, Videos, and Documents (12m)**
Step-by-step guide to sending different media types and managing file uploads in your bot.
- **Formatting Messages: Markdown, HTML, and Plain Text (11m)**
Make your bot's messages more readable and professional with text formatting options.
- **Managing User Sessions and Context (28m)**
Use ConversationHandler to manage multi-step interactions and store temporary user data.

- **Implementing a Simple FAQ or Help System (17m)**
Build a dynamic help system that answers common questions using a predefined structure.
- **Rate Limiting and Bot Security Basics (13m)**
Protect your bot from spam and abuse by implementing basic rate limiting and security checks.

Advanced Bot Functionality

- **Integrating External APIs with Your Bot (30m)**
Real case study analysis: Connect your bot to a weather API or news feed to provide dynamic responses.
- **Storing Data: Using Databases with SQLite or PostgreSQL (26m)**
Learn to save user preferences, logs, or any data persistently using a lightweight database.
- **Building a Polling or Voting Bot (24m)**
Create a bot that can create polls, collect votes, and display results in real-time.
- **Setting Up Webhooks for Real-Time Updates (32m)**
Move beyond polling by setting up a webhook for efficient, instant message processing.
- **Handling Payments and Subscriptions via Telegram (28m)**
A practical guide to integrating Telegram's payment API for bots that sell services or digital goods.
- **Creating a Bot for E-commerce: Order Tracking (35m)**
Step-by-step project to build a bot that tracks orders and provides status updates to customers.
- **Deploying with Docker for Scalability (22m)**
Containerize your bot for easy deployment and scaling across different environments.

Telegram Group Management Automation

- **Adding Your Bot to a Group and Setting Permissions (10m)**
Practical steps to add your bot to a group and configure admin rights correctly.
- **Listening to Group Messages and Events (18m)**
Learn how to capture new messages, member joins, and other group events in your bot.
- **Automating Welcome Messages for New Members (14m)**
Build a system to send personalized welcome messages and rules to every new user.
- **Implementing Anti-Spam and Auto-Moderation (26m)**
Create rules to automatically delete spam, restrict users, and keep the group clean.
- **Setting Up Automated Announcements and Pinned Messages (20m)**
Schedule regular announcements or manage pinned messages automatically from your bot.
- **Creating a Group FAQ or Knowledge Base Bot (25m)**
Develop a bot that answers common group questions on-demand, reducing admin workload.
- **Managing User Roles and Permissions (29m)**
Design a system to assign roles (e.g., moderator, VIP) and manage access within the group.
- **Running Polls and Surveys Within the Group (15m)**
Use your bot to create interactive polls and gather feedback from group members.
- **Handling Group Events: New Members, Left Members (17m)**
Code logic to respond to member joins and leaves for analytics or cleanup tasks.
- **Building a Group Activity Tracker (33m)**
Real case study: Create a bot that tracks user activity and provides engagement reports.

- **Integrating Group Bots with External Tools (Slack, Discord) (28m)**
Learn to sync Telegram group activity with other platforms for unified community management.
- **Deploying and Maintaining a Group Management Bot (21m)**
Best practices for hosting, monitoring, and updating your group bot for long-term reliability.

Pro-Level Projects and Best Practices

- **Project 1: Building a Customer Support Bot (40m)**
A comprehensive project to create a bot that handles support tickets, FAQs, and escalations.
- **Project 2: Creating a Content Distribution Bot (38m)**
Build a bot that automatically posts updates from your blog or social media to multiple channels.
- **Project 3: Developing a Multi-Group Admin Dashboard (36m)**
Design a dashboard to manage multiple Telegram groups from a single interface.
- **Testing Your Bot: Unit and Integration Testing (24m)**
Learn to write tests for your bot logic to ensure reliability before deployment.
- **Performance Optimization for High-Traffic Bots (31m)**
Techniques to handle thousands of users concurrently without slowing down.
- **Security Hardening: Protecting Your Bot and Data (27m)**
Advanced security measures, including encryption, secure storage, and access controls.
- **Monetizing Your Telegram Bot: Models and Examples (23m)**
Explore different revenue models, from subscriptions to affiliate marketing within your bot.
- **Marketing and Growing Your Bot's User Base (20m)**
Practical strategies to promote your bot and attract users organically.
- **Keeping Up with Telegram API Updates (12m)**
A guide on how to track and adapt to changes in the Telegram Bot API over time.
- **Case Study: A Successful Community Bot Analysis (29m)**
Deep dive into a real-world bot that grew a large community, analyzing its features and strategy.
- **Building a Bot with No-Code Tools (for Non-Developers) (16m)**
An overview of platforms like ManyChat or Botpress for rapid prototyping without coding.
- **Final Project Review and Code Refactoring (34m)**
A problem-solving session to review, improve, and structure your final bot project professionally.
- **Next Steps: Scaling, Partnerships, and Future Features (15m)**
Actionable roadmap for taking your bot to the next level, including collaboration and feature ideas.

Discover our blog

Learn more with free articles written by our professors and guest experts.

Latest article: **Flight Attendant Preparation: What to Study Before You Apply**

[Visit the blog](#) [All courses](#) [This course](#)

