

5 — Application System

Application System full: categories, voting, interviews, portal.

- [Enable the Application System](#)
- [Creating Categories](#)
- [Voting & Interviews](#)
- [Public Portal](#)

Enable the Application System

Enable the Application System

Sidebar → **Applications** → **Configuration**

Master toggle

At the top: "**Application system enabled**" switch. **ON**, otherwise nothing works.

Global settings

Setting	Default
Cooldown	7 days between applications per user
Min account age	30 days (user account must be that old)
Min server join	7 days (user must have been on the server that long)
Voting enabled	Yes
Voting channel	(Required if voting on)
Auto-expire	Applications without reply close after 14 days
Archive channel	(Optional) accepted/rejected applications go here
Interview reminder	30 min before interview time

Blacklist

User IDs who **cannot** apply.

123456789012345678

987654321098765432

Creating Categories

Application Categories

Sidebar → **Applications** → **Configuration** → **Categories**

A **category** is a position type, e.g. "Moderator", "Builder", "Streamer".

Per category

Field	Purpose
Name	What users see ("Moderator")
Emoji	(Optional) prefix in dropdown
Status	Open / Closed
Team role	Which role can vote
Accept roles	Which roles auto-assign on accept (multiple possible)
Form fields	Application questions (max 5 per category, or multi-modal for 5+)
Internal marker	(Bool) Internal-only

Multi-modal support

With **more than 5 questions**, Discord auto-shows a second modal. Lets you build complex applications with 10+ questions.

Status: Open / Closed

- **Open** — users can apply
- **Closed** — users see the position grayed out, can't click

Example category

Name: Moderator

Emoji: 🗳️

Status: Open

Team role: @Moderator

Accept role: @Moderator (same)

Form fields:

- How old are you? (number)
- Discord experience? (paragraph)
- Hours/week available? (number)
- Why you? (paragraph)

Voting & Interviews

Voting & Interview System

Voting

When **voting is enabled**, the bot posts an embed in the **voting channel** for each new application with:

- **Application content** (all answers)
- **Buttons:** Accept · Reject · Hold
- **Live counter** of who voted what
- **Dropdown menu** for detailed per-voter notes

Who votes?

Only members with the **team role of that category**.

What happens after voting?

You as the responsible person make the final call manually — via buttons in the application channel:

- **Accept** — user gets DM, accept-roles auto-assigned, application archived
- **Reject** — user gets DM, archived

Interviews

Schedule an interview

In the application inbox:

1. Open the application
2. Click "**Schedule interview**"
3. Enter **date + time**
4. Select **invited team members**

5. Save

What happens then?

- Applicant gets a **DM** with date/time
- 30 min before: **reminder** (DM to applicant + invitees, plus a ping in the application channel)
- You can open a voice channel for the interview

Reminder service

Runs every 5 minutes, checks all active interviews — server logs show debug output on send.

Public Portal

Public Application Portal

Quantix Tickets offers a **public portal** where your application positions are presented like job listings on a careers page.

Enable

Sidebar → **Applications** → **Configuration** → `portalEnabled: true`

Configuration

Setting	Purpose
Portal background image	Banner up top
Portal title	(Default: server name)
Portal description	Markdown beneath the title

URLs

```
Portal:    https://tickets.quantixbot.app/apply?server=<guildId>
Detail:    https://tickets.quantixbot.app/apply/job?server=<guildId>&job=<jobId>
Form:      https://tickets.quantixbot.app/apply/form?server=<guildId>&job=<jobId>
```

Stepper navigation

Detail pages show a stepper at the top:

```
Overview → Job → Application
```

Users see where they are.

Language

`?lang=en` or `?lang=de` in the URL → portal switches language.

Application form

Discord-OAuth login is required. The user fills the form and the backend channels the application into the normal application workflow (voting, interviews, etc.).

Server icon as header

The header automatically uses your Discord server icon and (if available) server banner as background.