

# Lokální chatovací server

Obchodní akademie, Vyšší odborná škola a Jazyková škola s právem státní jazykové zkoušky

Dokumentace závěrečného projektu OPS

Aleš Hanáček, Karla Šoustková, Klára Adámková

## 1. Úvod

Cílem projektu je vytvoření chatovacího serveru Rocket.Chat běžícím na operačním systému Ubuntu, který umožní komunikaci v reálném čase prostřednictvím textových zpráv mezi uživateli.

### 1.1 Využití v praxi

Rocket.Chat umožňuje zaměstnancům komunikovat mezi sebou prostřednictvím textových zpráv, hlasových zpráv a hovorů. Uživatelé mohou sdílet soubory a spolupracovat na projektech prostřednictvím tohoto serveru.

### 1.2 Použité materiály

- Oficiální dokumentace Rocket.Chat na platformě GitHub.
- **Diskuzní fóra:** pro řešení dotazů ohledně konfigurace jsme se obrátili na online komunitní fórum *stackoverflow.com*.
- **Knihovny a nástroje třetích stran:** využili jsme knihovny, jako jsou Node.js, MongoDB, Snap, Nginx, Git, GnuPG a GraphicsMagick.
- Youtube – [instalace MongoDB](#), [instalace Rocket.Chat](#).

### 1.3 Ověřitelné cíle

- Implementace základního serveru s možností připojení více uživatelů.
- Možnost odesílání textových zpráv mezi uživateli.
- Schopnost přihlášení, registrace a resetování hesla.
- Úspěšná instalace Rocket.Chat serveru v Ubuntu.
- Testování funkčnosti serveru v reálném provozu.

© 2006 The Authors  
 Journal compilation © 2006 Blackwell Publishing Ltd

## 2.1 Instalace MongoDB

Pro instalaci databáze MongoDB si potřebujete nainstalovat serverovou verzi Ubuntu na Váš počítač bez grafického prostředí. V příkazovém řádku zadejte následující příkazy:

```
sudo apt-get install GnuPG
```

→ Instalace knihovny GnuPG

```
curl -fsSL https://pgp.mongodb.com/server-6.0.asc | \sudo gpg -  
o/usr/share/keyrings/mongodb-server-6.0.gpg \ --dearmor
```

→ Importujte veřejný klíč MongoDB (GPG)

```
echo "deb [ arch=amd64,arm64 signed-by=/usr/share/keyrings/mongodb-  
server-6.0.gpg ] https://repo.mongodb.org/apt/ubuntu jammy/mongodb-  
org/6.0 multiverse" | sudo tee /etc/apt/sources.list.d/mongodb-org-6.0.list
```

```
sudo apt-get install mongodb-org
```

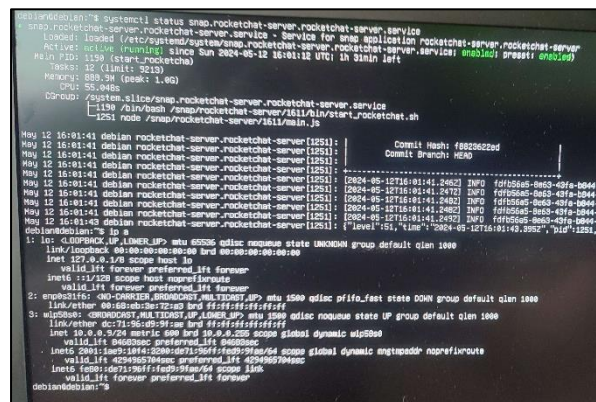
→ Instalujte balíčky MongoDB

```
sudo systemctl start mongod
```

```
sudo systemctl status mongod
```

```
sudo systemctl enable mongod
```

→ Zapněte a zkontrolujte status MongoDB



## 2.2 Instalace Rocket.Chat

Pro instalace Rocket.Chat serveru potřebujete tyto příkazy:

```
apt update && apt upgrade -y
```

→ Kontrola a instalace aktualizací systému

```
apt install nginx gnupg2 git unzip build-essential software-properties-common  
graphicsmagick
```

→ Instalace web serveru Nginx

## 2.3 Zprovoznění chatovací místnosti

Do našeho počítače, který běží na Ubuntu zadáme příkaz:

```
Ip a
```

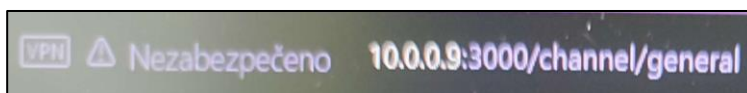
→ Zjištění ip adresy

Následně zadáme ip adresu + port na kterém běží Rocket.Chat do adresního řádku v prohlížeči druhého počítače. Nyní máme otevřenou stránku s chatovací místností.

Poznámka: server Rocket.Chat používá port 3000

Pro použití napište tedy do vyhledávání:

```
[vaše ip adresa]:3000
```

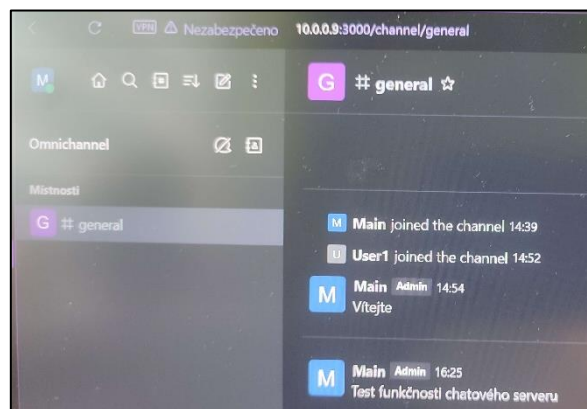


### 3. Dokumentace testování/ověření cílů

#### 1. Implementace základního serveru s možností připojení více uživatelů:

**Provedené testování:** Spuštění Rocket.Chat serveru na lokální síti s připojením dvou klientů na různých zařízeních. Každý klient měl svůj účet a byl schopen vidět zprávy odeslané druhým uživatelem.

**Výsledek testování:** Základní funkčnost serveru byla úspěšně ověřena.



#### 2. Možnost odesílání textových zpráv mezi uživateli:

**Provedené testování:** Odeslání textové zprávy z jednoho klienta a ověření, že ji druhý klient úspěšně obdrží.

**Výsledek testování:** Možnost odesílání textových zpráv mezi uživateli byla úspěšně ověřena.

#### 3. Schopnost přihlášení, registrace a resetování hesla:

**Provedené testování:** Vytvoření uživatelského účtu prostřednictvím Rocket.Chat cloudu a ověření, že je možné úspěšně přihlásit se k Rocket.Chat serveru s vytvořeným účtem.

**Výsledek testování:** Schopnost přihlášení a registrace uživatelů byla úspěšně ověřena. Resetování hesla nebylo testováno. Registrace uživatelského účtu probíhá prostřednictvím *Rocket.Chat cloudu*, kde uživatelé vyplní registrační formulář a vytvoří svůj účet.

#### 4. Úspěšná instalace Rocket.Chat serveru v Ubuntu:

**Provedené testování:** Instalace Rocket.Chat serveru v operačním systému Ubuntu.

**Výsledek testování:** Instalace Rocket.Chat serveru byla úspěšně provedena.

#### 5. Testování funkčnosti serveru v reálném provozu:

**Provedené testování:** Spuštění Rocket.Chat serveru na domácí lokální síti. Testování stability, rychlosti a spolehlivosti odesílání zpráv.

**Výsledek testování:** Server byl úspěšně spuštěn na domácí síti a jeho funkčnost byla ověřena. Rychlost odesílání zpráv byla optimální.

#### 4. Rozdělení práce

**Aleš Hanáček** – konfigurace, instalace a nastavení Rocket.Chat na serveru, připojení klientů. Zaznamenání případných chyb.

**Karla Šoustková** – dokumentace celého procesu, instalace potřebných knihoven a testování s více možnostmi (vyzkoušení různých databází)

**Klára Adámková** – testování serveru (ověření správné funkčnosti komunikace), hledání materiálu/zdrojů ze kterých jsme čerpali.

#### 5. Závěr

##### 1. Které části projektu proběhly úspěšně/neúspěšně

Povedlo se úspěšně nasadit a provozovat Rocket.Chat server na operačním systému Ubuntu. Funkčnost serveru byla ověřena a potvrzena. Nepovedlo se však úspěšně provést instalaci na operačním systému Debian.

---

## 2. Části projektu, které nás překvapily

Zaskočilo nás zjištění rozdílné kvality stability mezi operačními systémy Ubuntu a Debian.

## 3. Varianty řešení (výhody a nevýhody)

Jako alternativní řešení může být instalace Rocket.Chat serveru na jiných operačních systémech, které by mohly poskytnout lepší stabilitu a kompatibilitu s Rocket.Chat serverem.

## 4. Na co by si měli dát pozor Ti, kdo budou projekt zkoušet podle našeho návodu:

Při zkoušení projektu podle našeho návodu je důležité si pamatovat rozdíly mezi operačními systémy. Doporučujeme neprovádět instalaci prostřednictvím databáze MongoDB kvůli potenciálním problémům. Namísto toho je lepší zvolit alternativní databázové řešení, jako je například FerretDB, které může poskytnout stabilnější provoz.

## 5. Jak by se mohlo pokračovat

Pro další vylepšení projektu by bylo vhodné prozkoumat možnosti vytvoření veřejně dostupného serveru, který by umožnil komunikaci mezi uživateli.