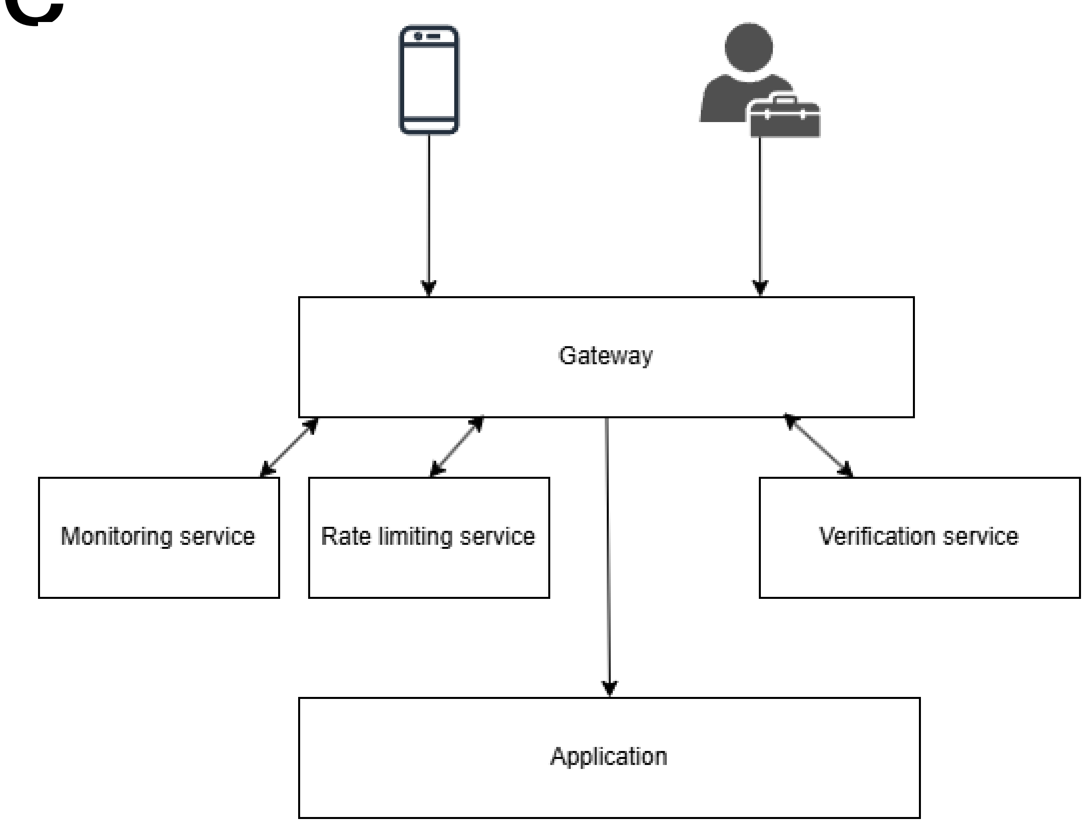
SECURE PROGRAMMING COURSEWORK

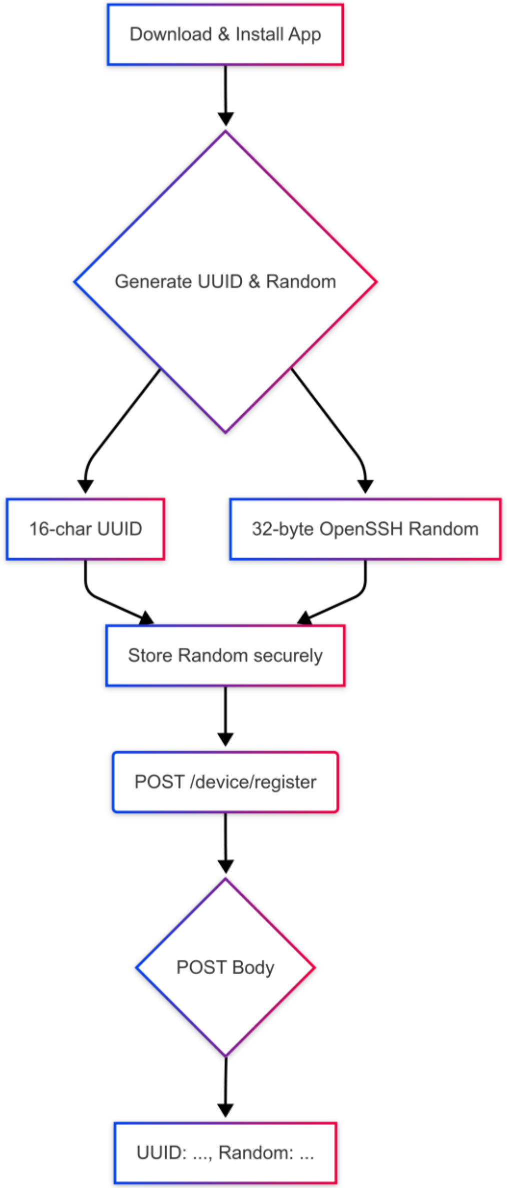
Essi Varrela

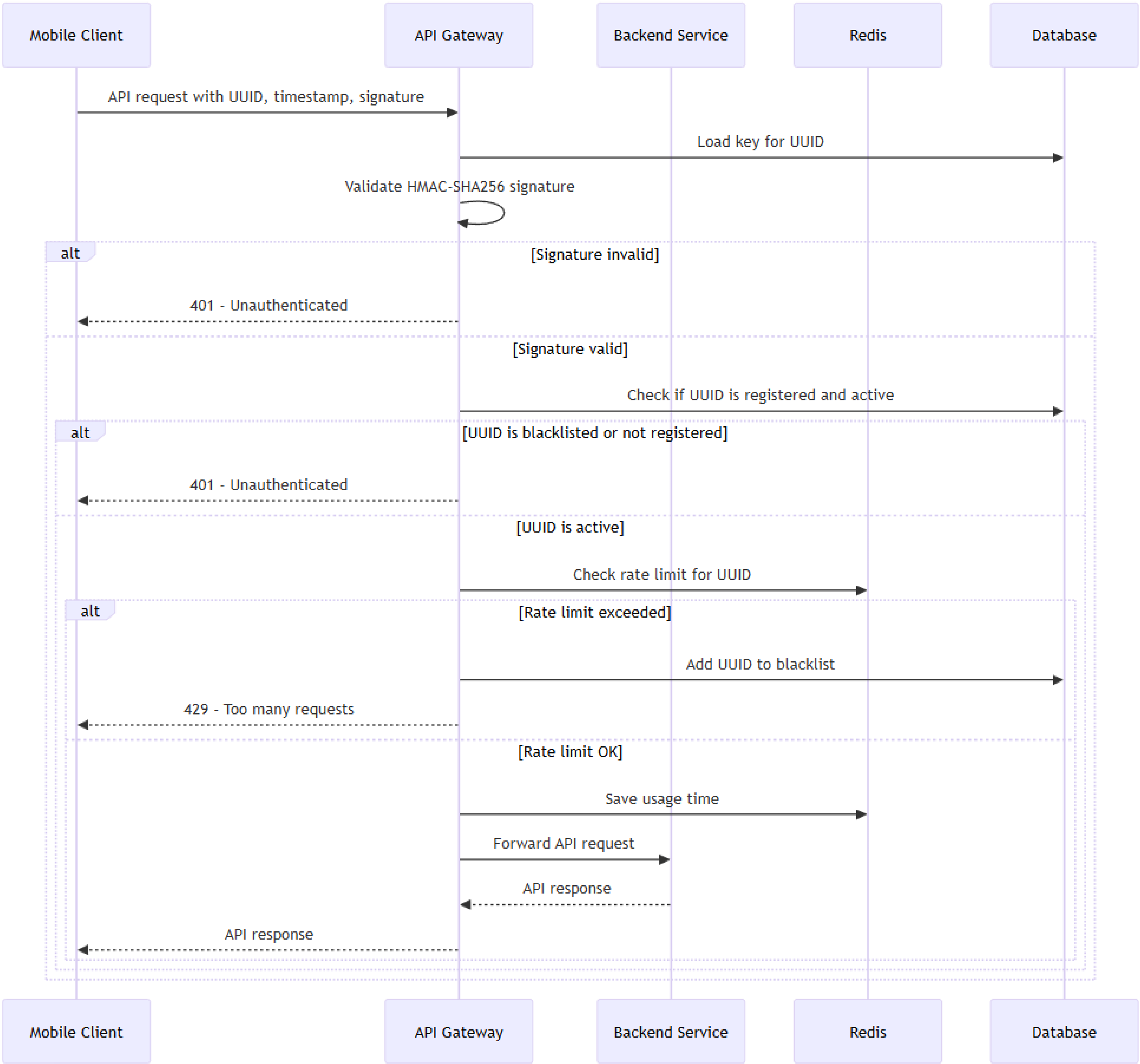
Gateway for mobile clients

- No login required
- Security relies on
  - Validating authenticity
  - Rate limiting
  - Blacklists
  - Monitoring

Using Java & Spring Boot




Download & Install App
Generate UUID & Random
16-char UUID
32-byte OpenSSH Random
Store Random securely
POST /device/register
POST Body
UUID: ..., Random: ...


Mobile Client
API Gateway
Backend Service
Redis
Database
API request with UUID, timestamp, signature
Load key for UUID
Validate HMAC-SHA256 signature
alt
[Signature invalid]
401 - Unauthenticated
[Signature valid]
Check if UUID is registered and active
alt
[UUID is blacklisted or not registered]
401 - Unauthenticated
[UUID is active]
Check rate limit for UUID
alt
[Rate limit exceeded]
Add UUID to blacklist
429 - Too many requests
[Rate limit OK]
Save usage time
Forward API request
API response
API response
Mobile Client
API Gateway
Backend Service
Redis
Database

Security features

- Devices register with UUID + secret key
  - Secret key is stored encrypted (AES)
  - These are expected to be stored securely in the client side for future requests to API
- Requests from mobile client are HMAC-SHA256 signed.
- Authenticity is verified in each request.
- Devices must be registered and activated for the request to succeed.
- UUID and IP-address are used for rate limiting
- Clients who violate the rules can be automatically blacklisted
  - Currently too many requests set the client to the blacklist

OWASP API Top 10 was used as a checklist and development was done following Secure Programming Practices

Monitoring & Alerts

- Custom logging filter for detecting cyber attacks.
- Prometheus is used to collect metrics.
- Grafana dashboard can be used to monitor:
  - Unusual usage times
  - Too many and Unauthenticated responses
- Alerts can be added from Grafana

Testing

- Unit + Integration tests (JUnit, Mockito, Spring Boot Test)
- Static Analysis (SAST): CodeQL
  - No findings
- Dynamic application security testing (DAST): OWASP ZAP
  - 2 Low findings
- Trivy: misconfiguration scanning
  - 2 findings
- CI pipeline (GitHub Actions) builds, tests, and scans

Future improvements

- Add authentication & authorization for admin endpoints
- Add dependency vulnerability checks to CI
    - Add also other tests to CI-pipeline – by solving the problem with multi module project first.
- Force periodic client secret rotation?