INTRODUCTION TO SOFTWARE BUSINESS PRODUCT MANAGEMENT

Week 4 Day 2

Led by: Emily Crose

for

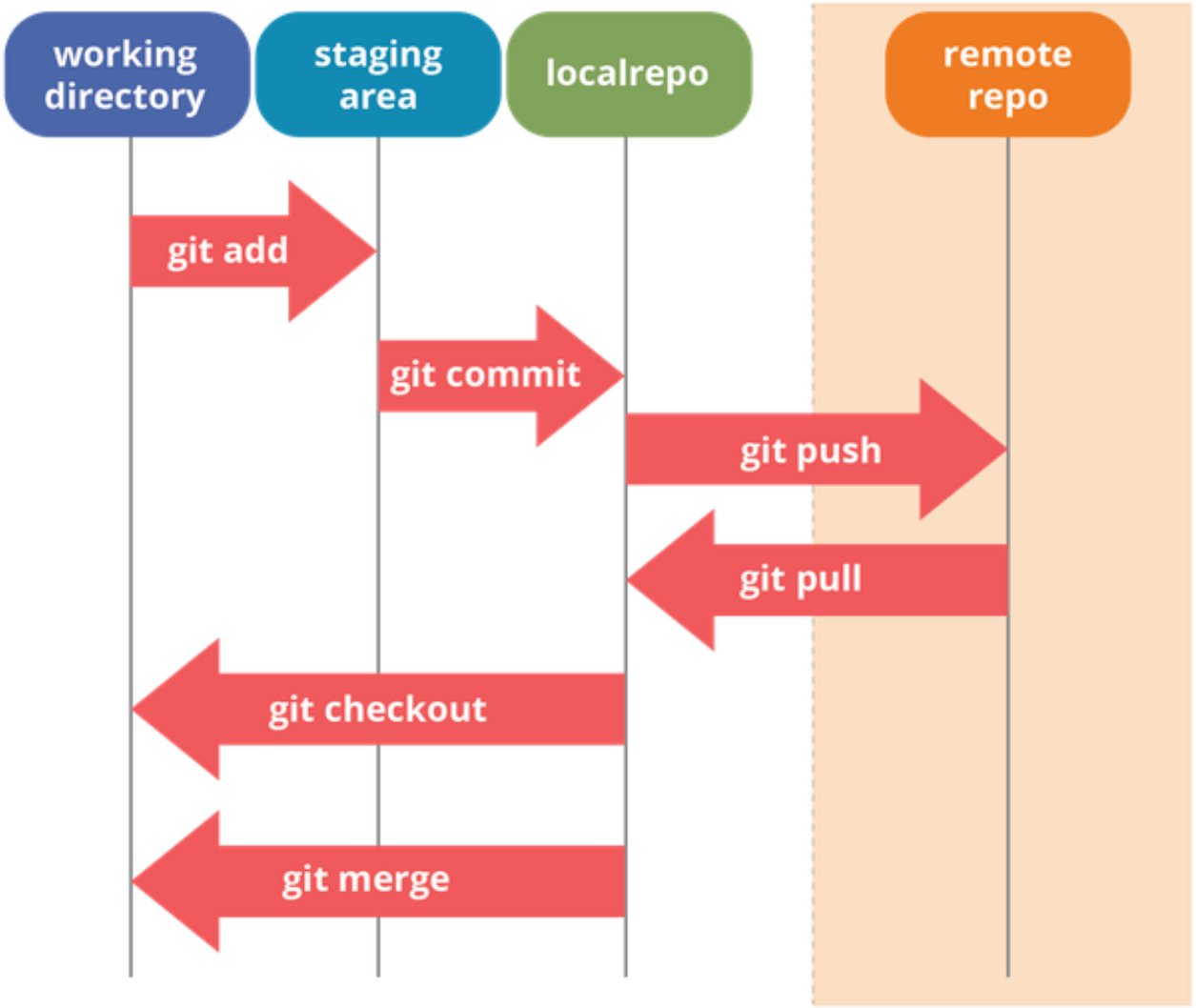
Oakland University

REVIEW OF DAY 1

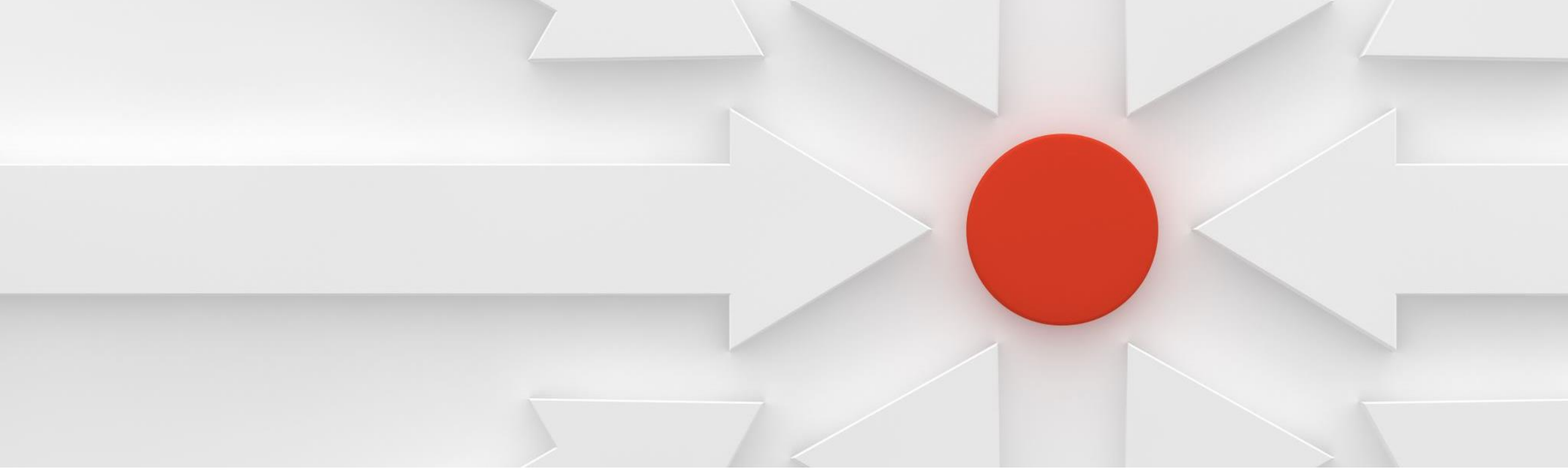
QUESTIONS FROM DAY 1?

PRACTICING WITH GIT

- https://learngitbranching.js.org/
- Alternatives
  - https://gitimmersion.com/lab_01.html



WORKFLOW REVISITED



BRANCHING WITH TEAMS

BRANCH NAMING CONVENTIONS

- What type of branch naming convention works for you?
- [Username] - <feature>
- [Jira ticket #] - <feature>
- [github issue number] – {username} - <feature>

PULL

FORKING

EDITING A FORKED REPOSITORY

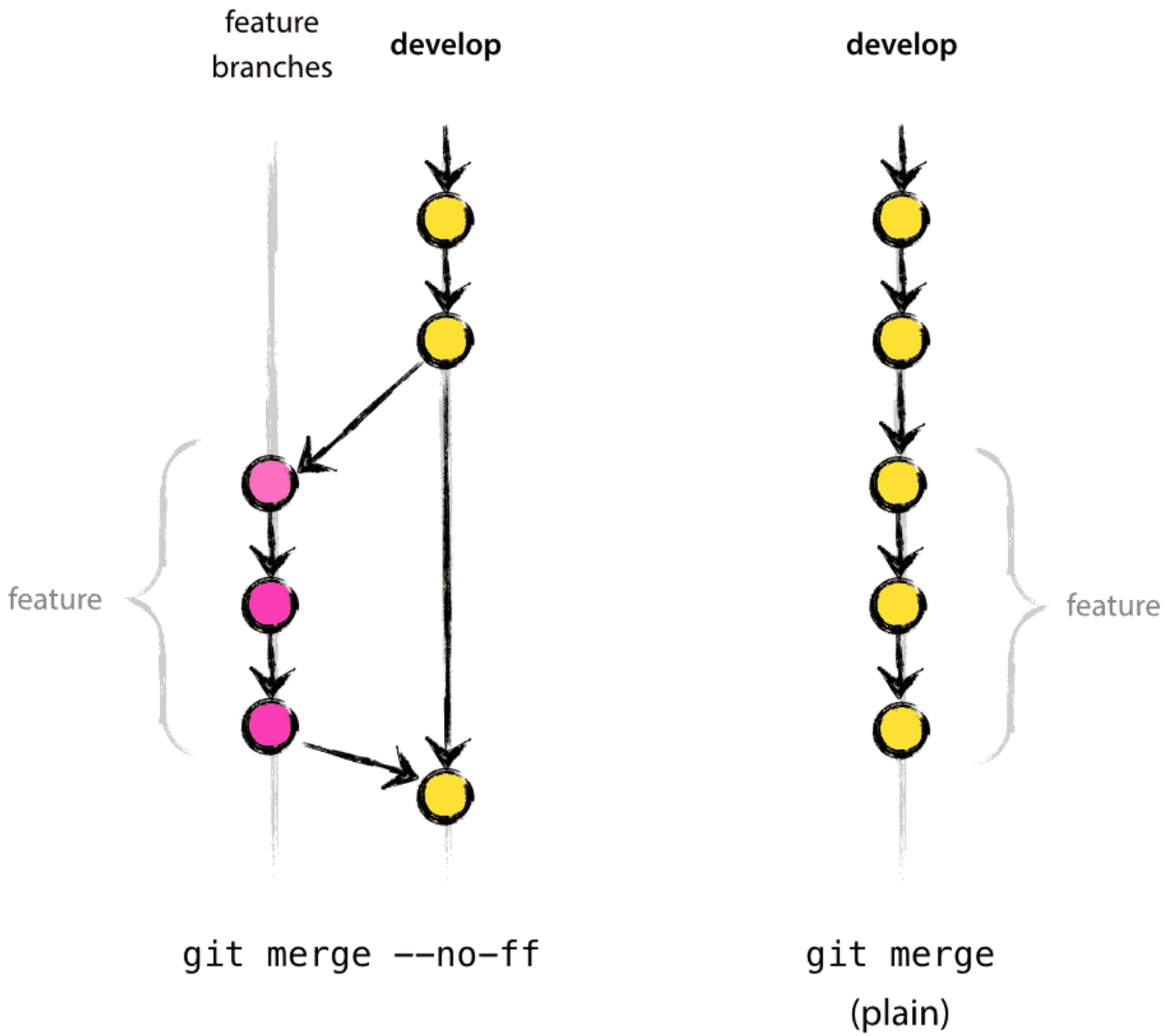
PULL REQUEST

CODE REVIEWS

CODE REVIEW

- Allows changes to be approved
- Limits bad code merges
- Adds a layer of safety

MERGING CHANGES



RELEASE CANDIDATES

BUILDING BINARIES

RELEASE NOTES

LICENSING

LICENSING

- Software can be copyrighted
- How do we protect our “Intellectual Property”?

BROAD CATEGORIES OF SOFTWARE LICENSING

- Weak Copyleft
- Copyleft
- Commercial or Proprietary
- Dual
- Public Domain

PERMISSIVE

- Permissive
  - Minimally restrictive on modifications or redistribution
  - Typically, only require acknowledgements

WEAK COPYLEFT

- Allows linking to open source libraries
- Minimal requirements
- Modifying the library is more complicated than permissive
- Examples:
  - Gnu Lesser General Public License (GLPL)
  - Mozilla Public License (MPL)
  - CDDL
  - Eclipse

Free as in Freedom

COPYLEFT

- A.K.A. “Reciprocal” licenses or “Restrictive” licenses
- Not as commercially friendly
- Requires publication of source code for derivative works
  - Not good for commercial products!
- Examples:
  - Gnu Public License (GPL)

10 MINUTE BREAK

COMMERCIAL/PROPRIETARY

- Most restrictive
- Typically used by corporations
- Typically closed source

DUAL

- Differs for different types of users
- Examples:
  - Server-Side Public License (SSPL)

PUBLIC DOMAIN

- Copyright protections do not apply
- Hard to find software in this category
- Most permissive

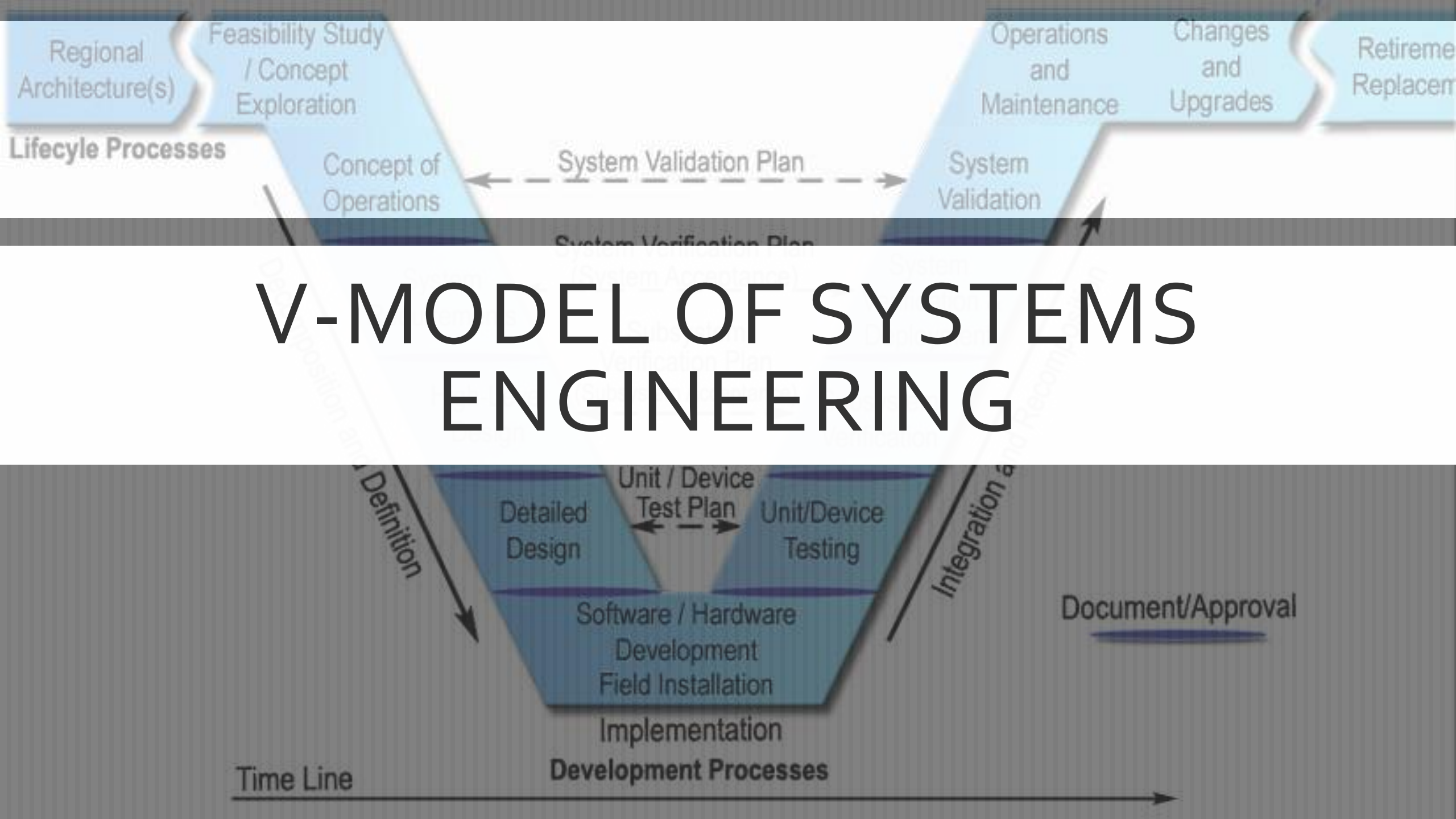
UNLICENSED

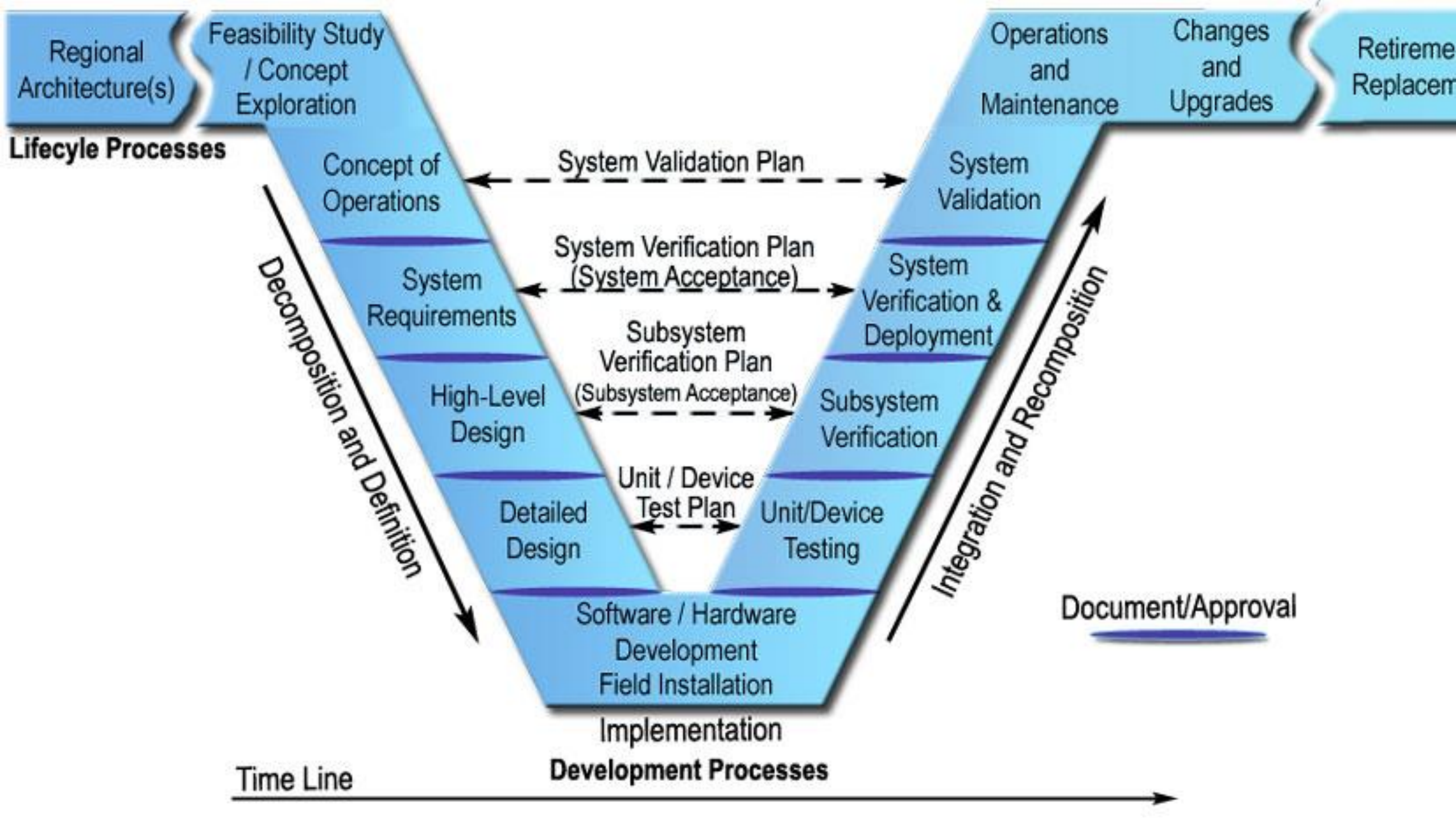
Can be complicated

LICENSING CONCERNS

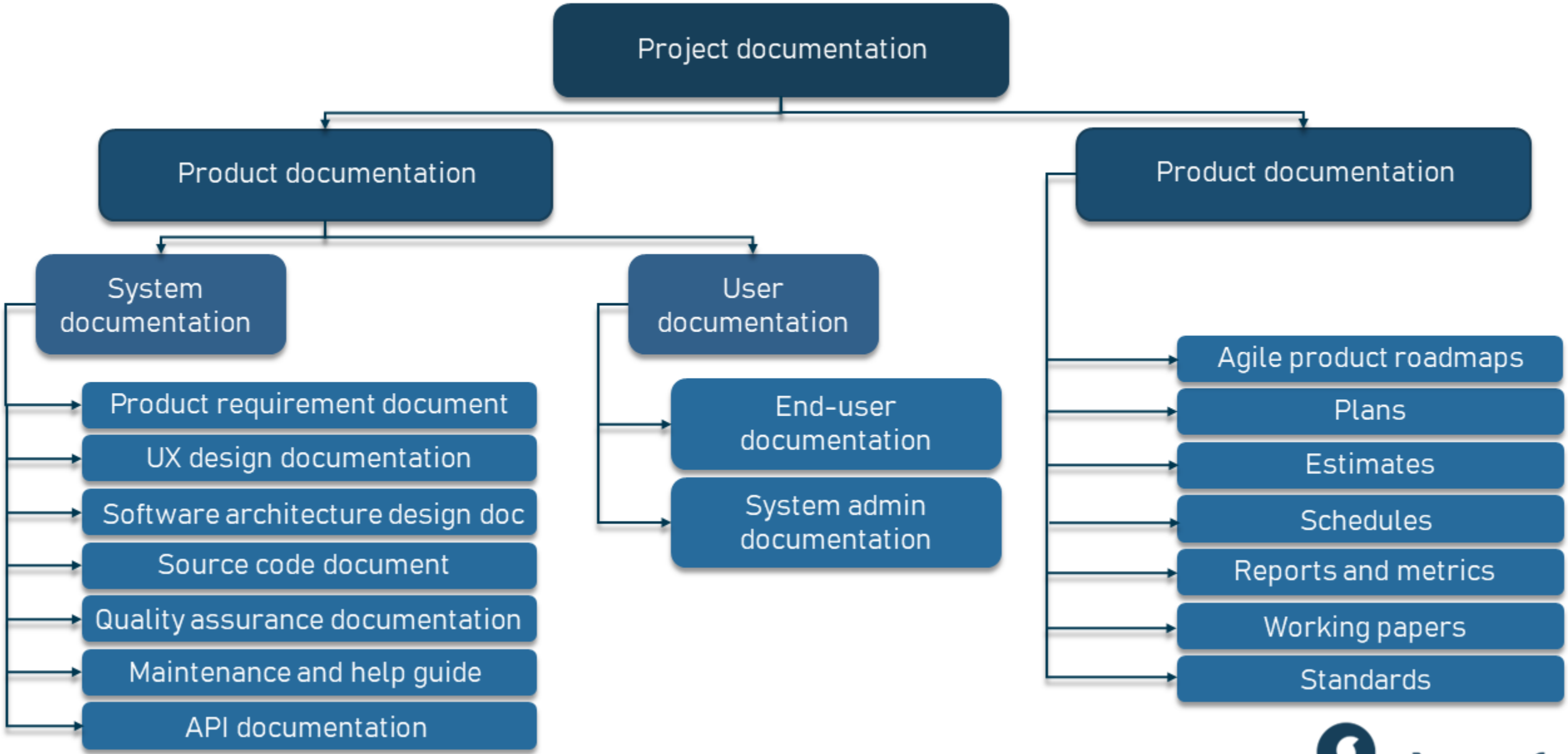
ARCHITECTURE DESIGN

V-MODEL OF SYSTEMS ENGINEERING




Regional Architecture(s)
Feasibility Study / Concept Exploration
Operations and Maintenance
Changes and Upgrades
Retireme Replacem
Lifecyle Processes
Concept of Operations
System Validation Plan
System Validation
System Requirements
System Verification Plan (System Acceptance)
System Verification & Deployment
Decomposition and Definition
Subsystem Verification Plan (Subsystem Acceptance)
High-Level Design
Subsystem Verification
Unit / Device Test Plan
Detailed Design
Unit/Device Testing
Integration and Recomposition
Document/Approval
Software / Hardware Development Field Installation
Implementation
Development Processes
Time Line

SOFTWARE DOCUMENTATION TYPES



altexsoft

Tech Pillar 3 – WP4
Augmented Intelligence

Tech Pillar 2 – WP5
Augmented Reality

Tech Pillar 1 – WP3
Augmented Communication & Sensing

W3C RDF
W3C SPARQL
W3C OWL

Swarm Intelligence & Mission Planning
STANAG 4586 - UAV
Decision Support
Task/Resource/Incident Management - STANAG 5525 / JC3IEDM

based on Frontex CIRAM
Communication
Visual recognition
Satellite Communication
Sensor Fusion
Simulation environment

ARESIBO Integration Engine

Interoperability of Networks

Interoperability of Networks

C2 AR Interfaces

Serious games

Interoperability of Interfaces

Field AR Interfaces

GSM / 4G / LTE / Satcom

Hybrid Intelligent Network

GSM / 4G / LTE / Satcom

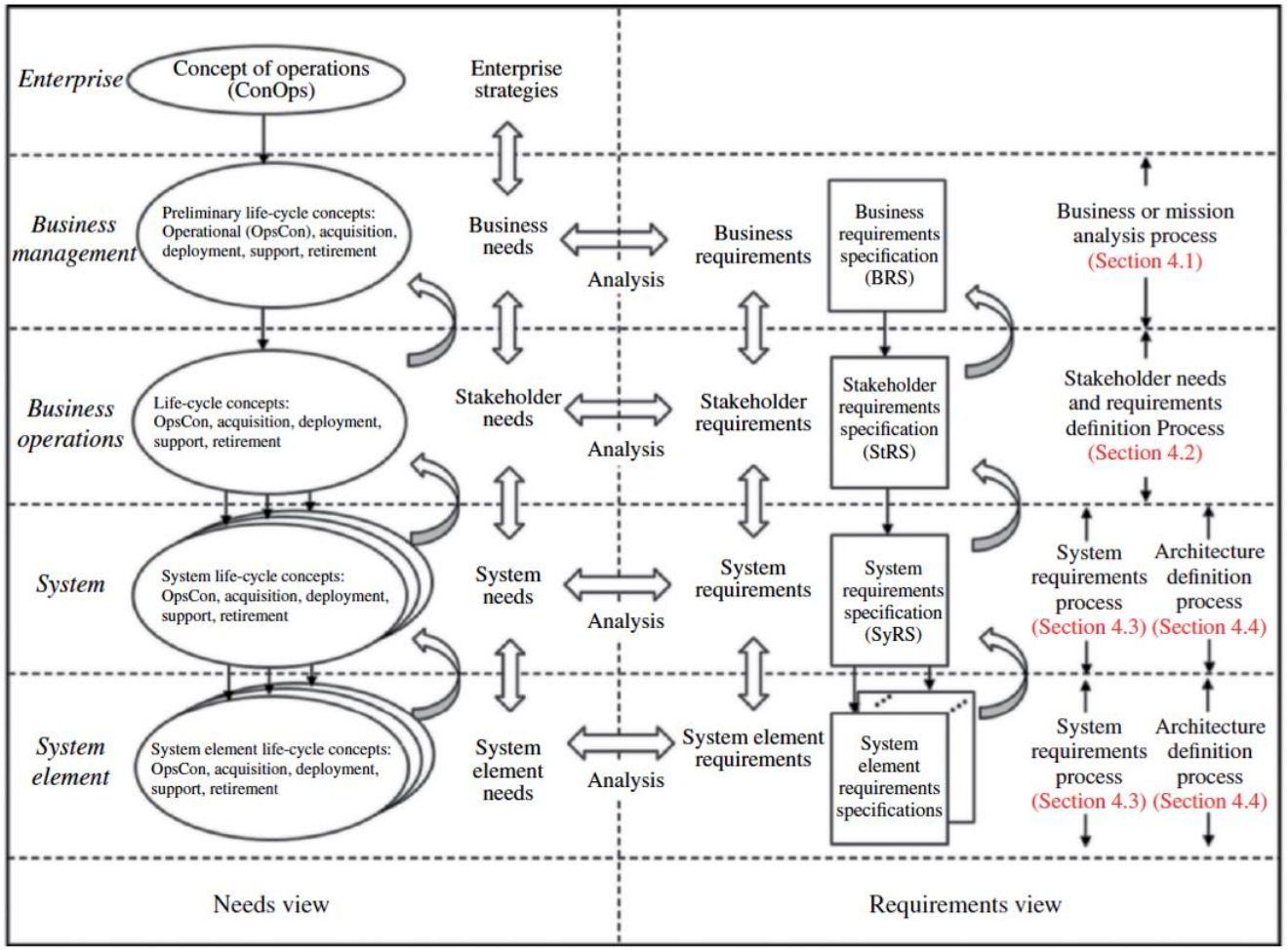
Edge Computing

Maritime Borders –

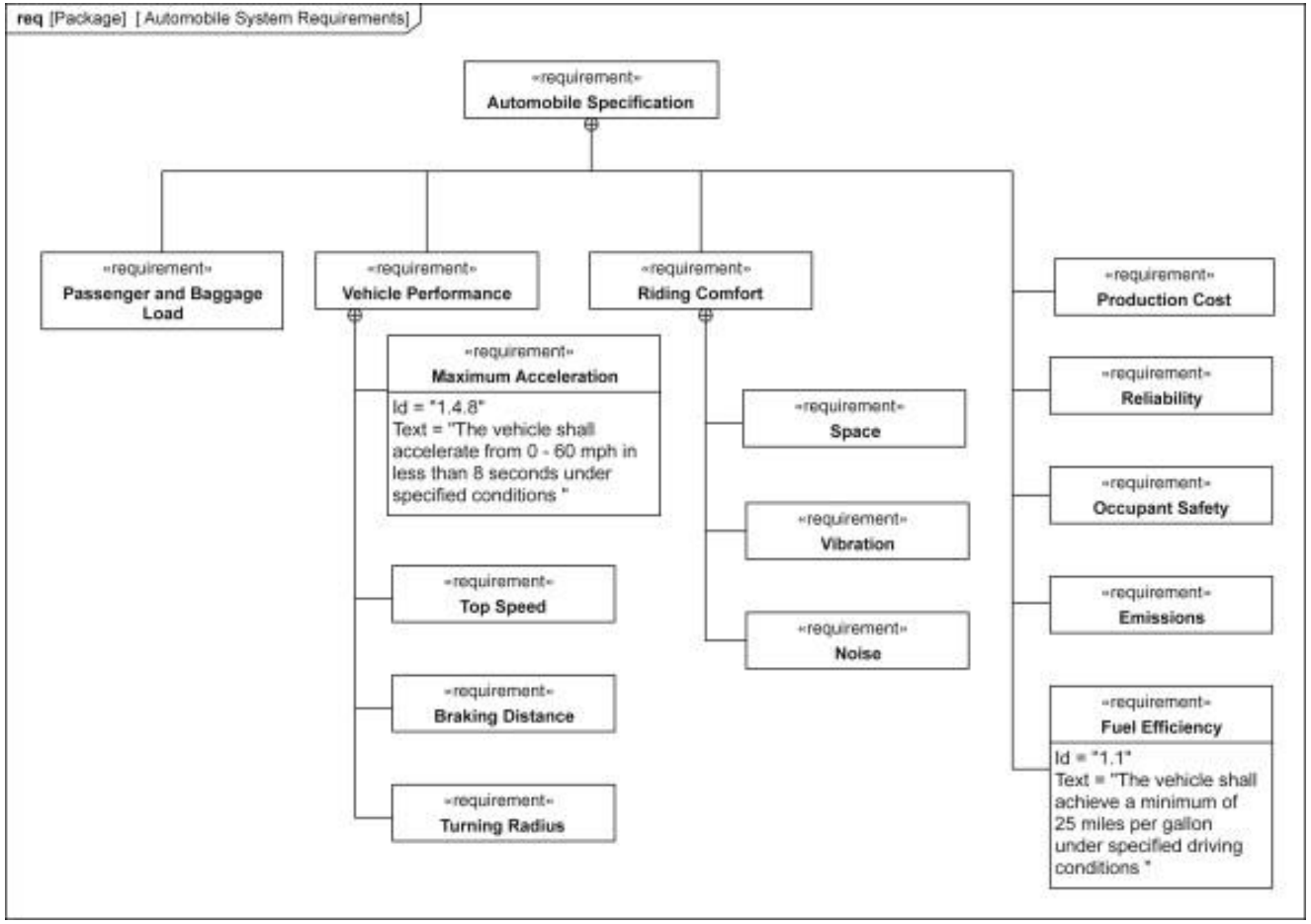
Underwater Acoustic

Man-Machine (M2)
Collaboratio

Land Borders –

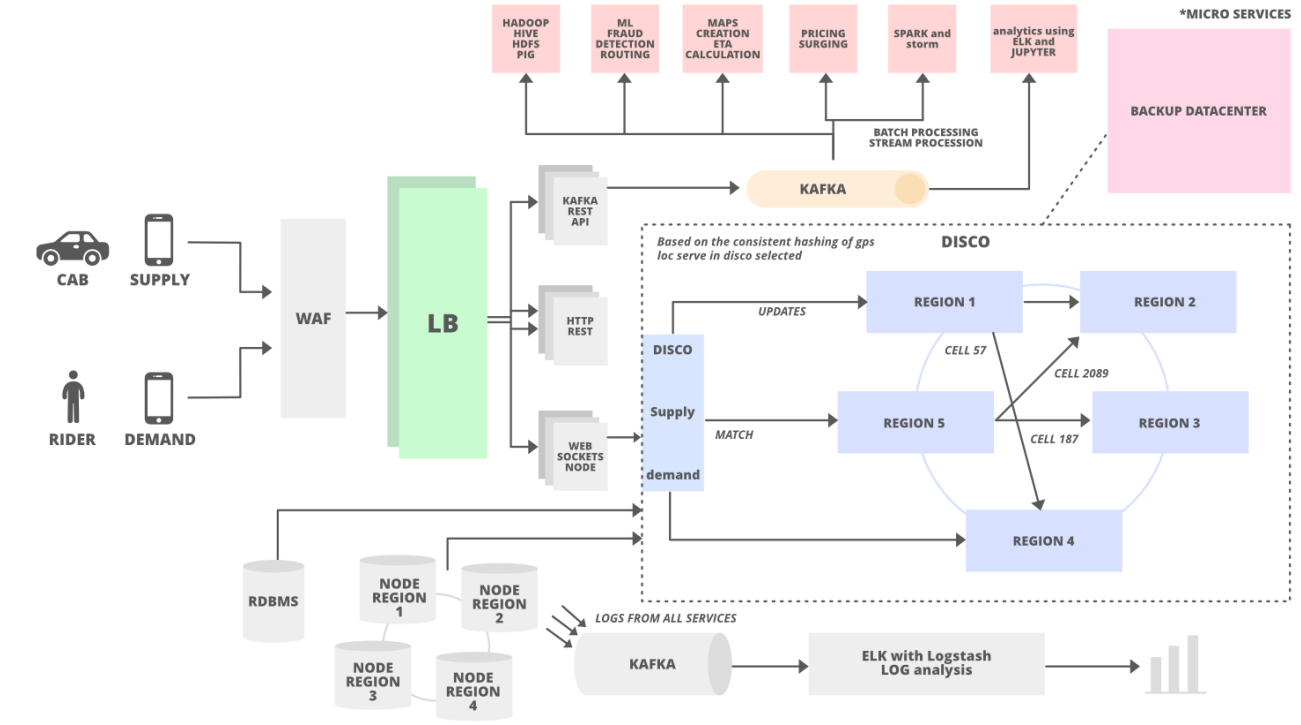


CONCEPT OF OPERATIONS (CONOP)



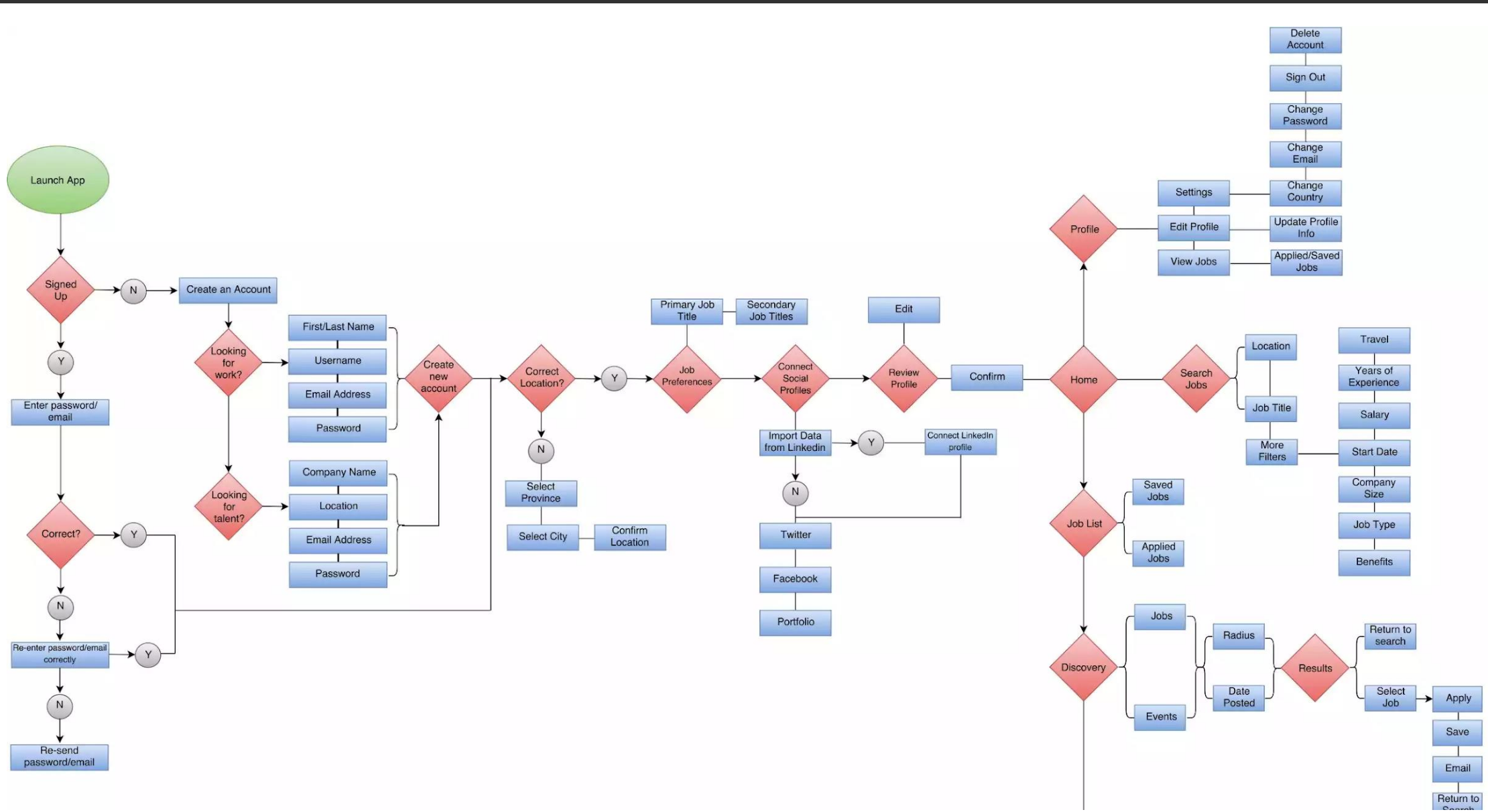
SYSTEM REQUIREMENTS DIAGRAM

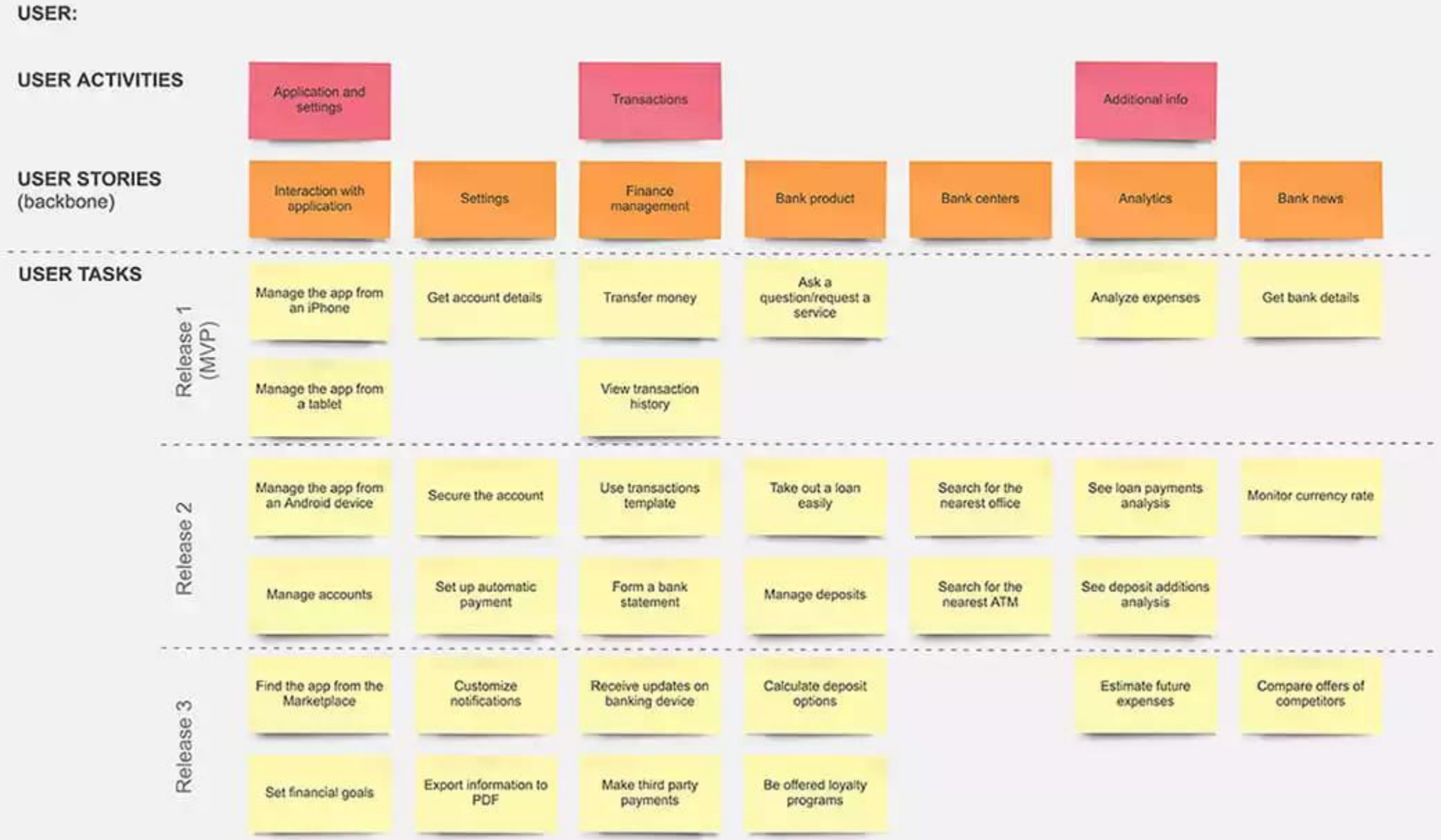
HIGH LEVEL DESIGN DOCUMENTATION

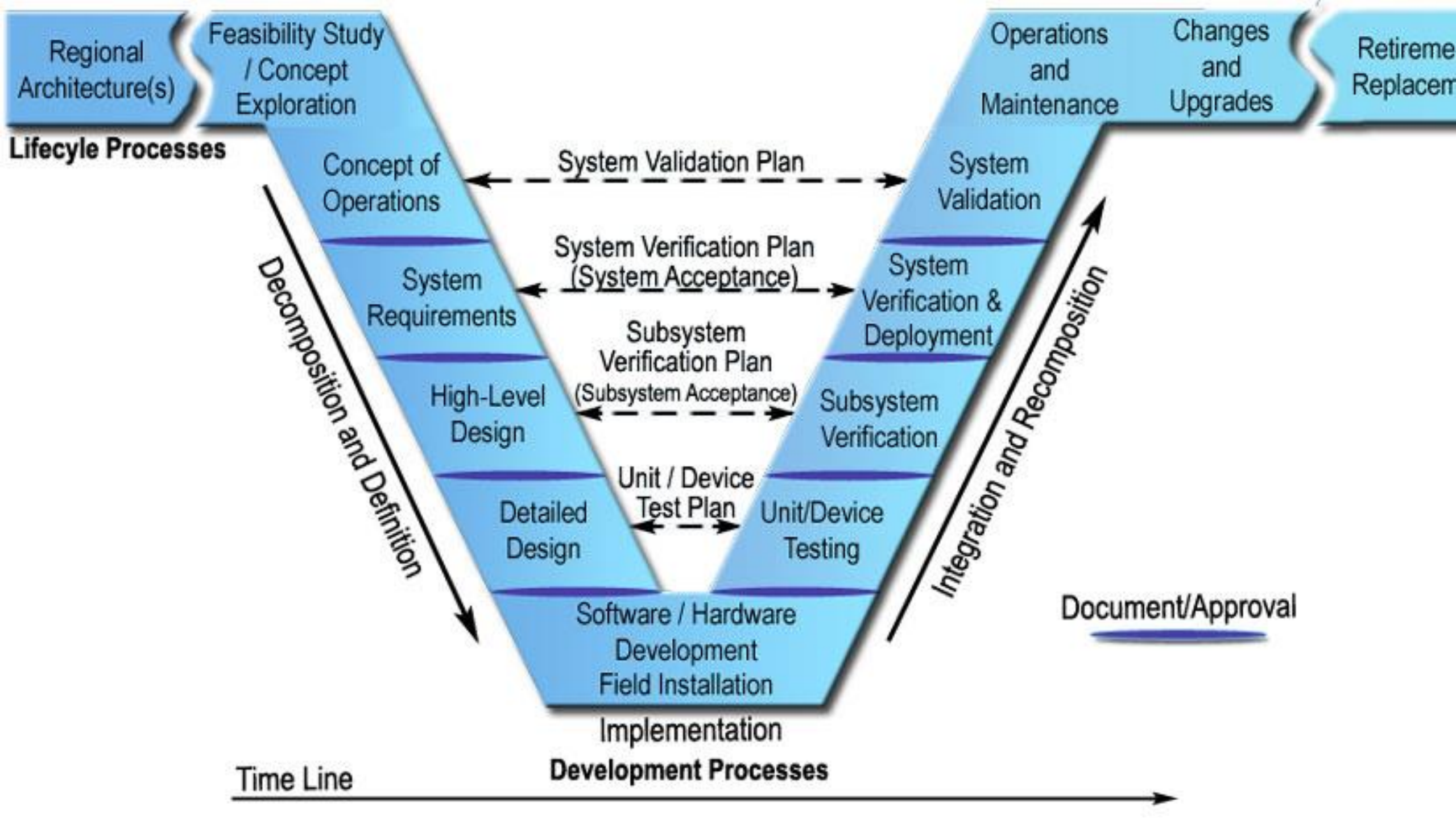


DETAILED ARCHITECTURE DESIGN DOCUMENTATION

- User flow diagram
- Site/product map
- User story map


Launch App
Signed Up
N
Create an Account
Looking for work?
First/Last Name
Username
Email Address
Password
Looking for talent?
Company Name
Location
Email Address
Password
Create new account
Y
Enter password/ email
Correct?
Y
N
Re-enter password/email correctly
Y
N
Re-send password/email
Correct Location?
Y
N
Select Province
Select City
Confirm Location
Primary Job Title
Secondary Job Titles
Job Preferences
Connect Social Profiles
Import Data from Linkedin
Y
Connect LinkedIn profile
N
Twitter
Facebook
Portfolio
Edit
Review Profile
Confirm
Home
Profile
Settings
Edit Profile
View Jobs
Delete Account
Sign Out
Change Password
Change Email
Change Country
Update Profile Info
Applied/Saved Jobs
Search Jobs
Location
Job Title
More Filters
Travel
Years of Experience
Salary
Start Date
Company Size
Job Type
Benefits
Job List
Saved Jobs
Applied Jobs
Discovery
Jobs
Events
Radius
Date Posted
Results
Return to search
Select Job
Apply
Save
Email
Return to


USER:
USER ACTIVITIES
Application and settings
Transactions
Additional info
USER STORIES
(backbone)
Interaction with application
Settings
Finance management
Bank product
Bank centers
Analytics
Bank news
USER TASKS
Release 1 (MVP)
Manage the app from an iPhone
Get account details
Transfer money
Ask a question/request a service
Analyze expenses
Get bank details
Manage the app from a tablet
View transaction history
Release 2
Manage the app from an Android device
Secure the account
Use transactions template
Take out a loan easily
Search for the nearest office
See loan payments analysis
Monitor currency rate
Manage accounts
Set up automatic payment
Form a bank statement
Manage deposits
Search for the nearest ATM
See deposit additions analysis
Release 3
Find the app from the Marketplace
Customize notifications
Receive updates on banking device
Calculate deposit options
Estimate future expenses
Compare offers of competitors
Set financial goals
Export information to PDF
Make third party payments
Be offered loyalty programs


Regional Architecture(s)
Feasibility Study / Concept Exploration
Operations and Maintenance
Changes and Upgrades
Retireme Replacem
Lifecyle Processes
Concept of Operations
System Validation Plan
System Validation
System Requirements
System Verification Plan (System Acceptance)
System Verification & Deployment
Decomposition and Definition
Subsystem Verification Plan (Subsystem Acceptance)
High-Level Design
Subsystem Verification
Unit / Device Test Plan
Detailed Design
Unit/Device Testing
Integration and Recomposition
Document/Approval
Software / Hardware Development Field Installation
Implementation
Development Processes
Time Line

OPERATIONS & MAINTENANCE

AUTOMATING OPERATIONS

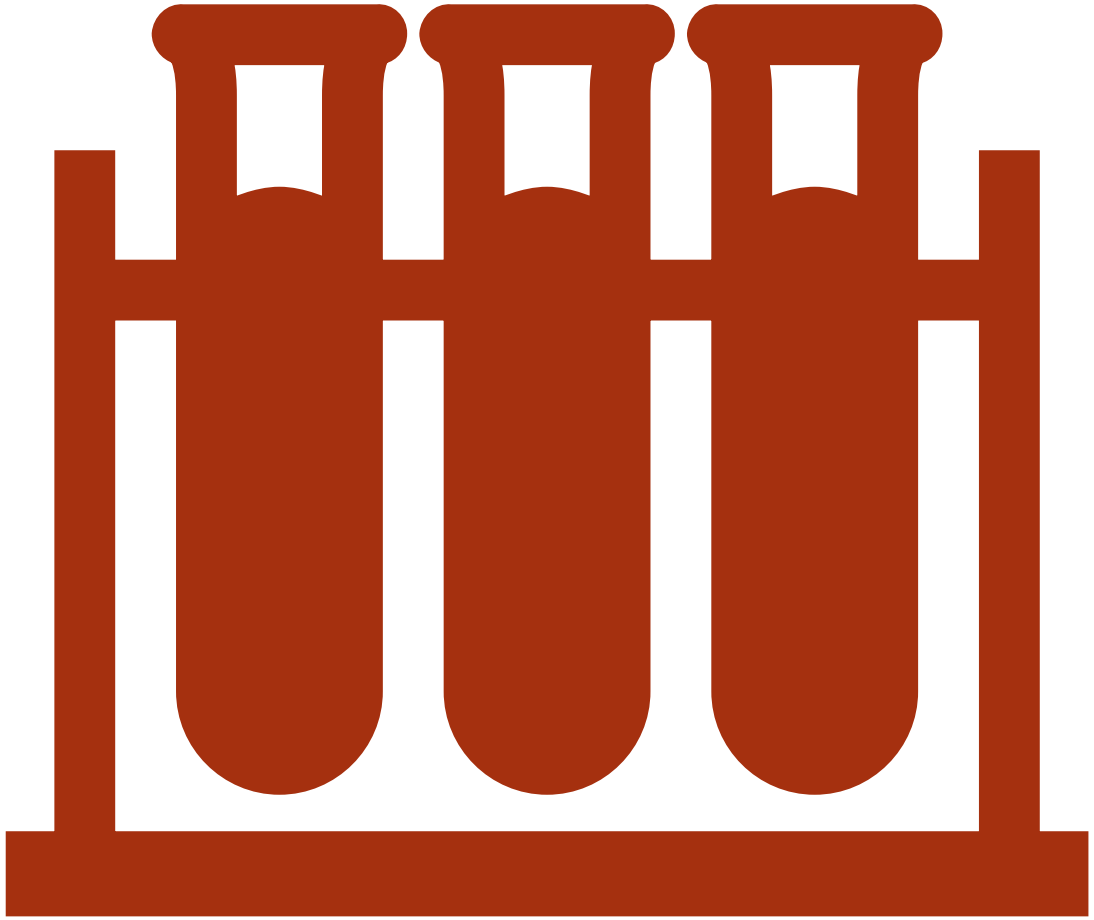
- Environment/Deployment Automation
  - Jenkins
  - Chef
  - Ansible
- Pipeline building
- Changes & upgrades
  - Can be easily done with automation scripting!

RETIREMENT & REPLACEMENT

END OF LIFE CONSIDERATIONS

- Can we actually afford to retire this system?
- What will replace it?
- Will this be transparent to users?
- Where will new users go?

10 MINUTE BREAK



FUNCTIONAL TESTING

MANUAL TESTING?

Takes longer

Not scalable

Tedious!

AUTOMATED TESTING

What does our testing pipeline look like from an architecture perspective?

What types of tests do we need to perform?

Regression testing

QA testing

User Acceptance Testing

Load testing

Performance Testing

Security Testing

TESTING IN DEPTH

LOAD TESTING

STRESS TESTING

PERFORMANCE TESTING

SECURITY TESTING

UNIT TESTING

- We added a feature!
  - Does it do what it's supposed to do?
- This should be done for every new feature!

LOAD TESTING

Average number of concurrent users in the tool at a time

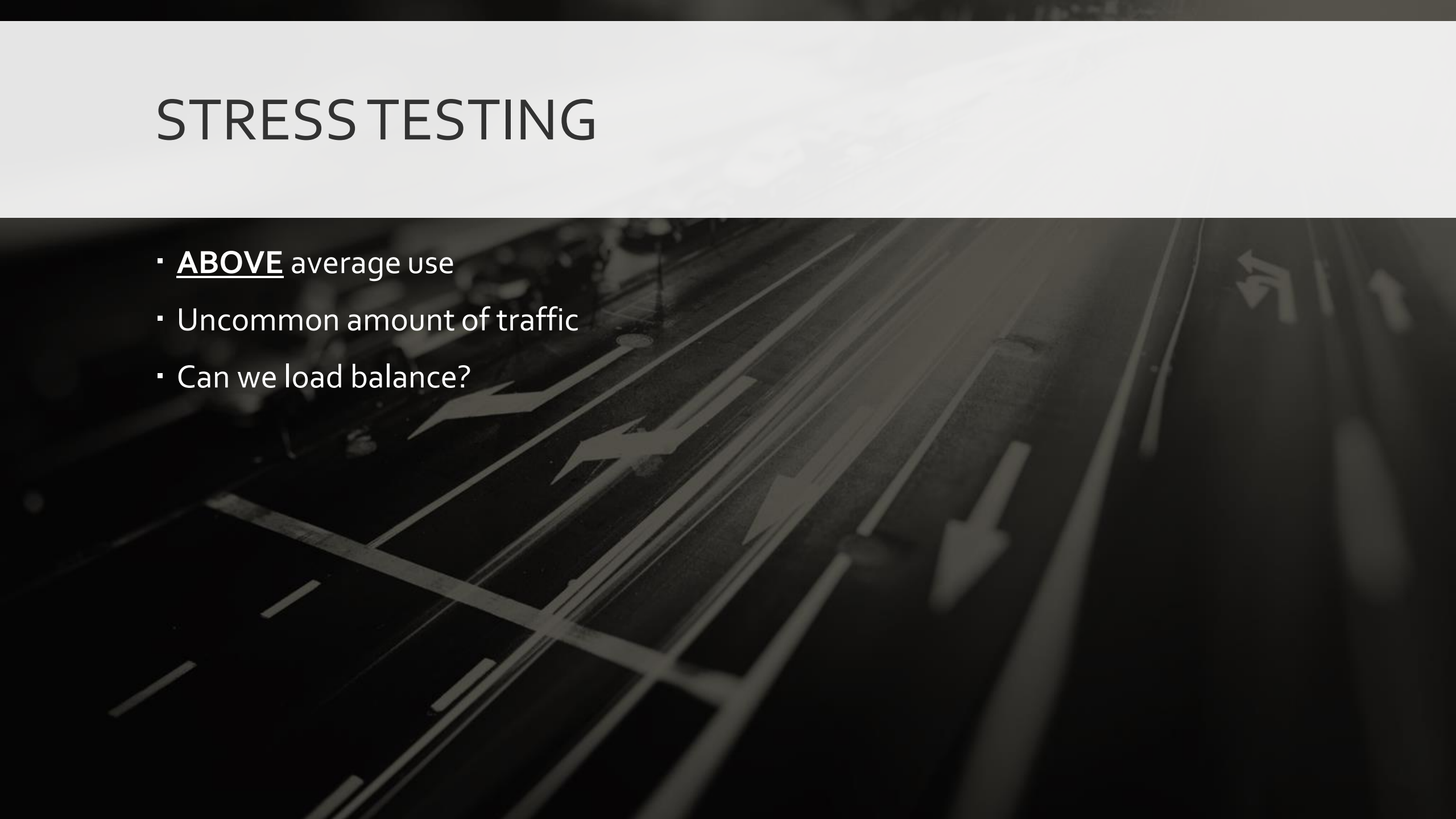
Beneath “peak” use

This is the **REGULAR** amount of traffic you expect to get

Does our app scale with more users?

We can test for this!

STRESS TESTING



- **ABOVE** average use
- Uncommon amount of traffic
- Can we load balance?

PERFORMANCE TESTING

- How quickly can we load information?
- Do we see any lags in transfer under regular use?
  - Lags in higher-than-average use?

SECURITY TESTING

- How can we break this app?
  - Can we get access we shouldn't have?
  - Can we get information we shouldn't have?
  - Can we make the app do something unexpected?
- Do we need to fully pentest this app?

PENTESTING?

- Can we do this “in-house”?
- Is there a cost associated with this?
  - Probably!
  - Is doing a pentest worth the cost?
- What if we don’t do a pentest?

PRODUCT PENTESTING OPTIONS

- In-house pentest
  - Do we have the expertise to do this in-house?
- Third-party testing
  - Is there a legal or regulatory reason we should do a third-party pentest?

ARCHITECTURE PENTESTING

- Should we pentest our back-end architecture?
  - Is there a cost associated with this?
    - Probably!
-

OUTCOMES FROM TESTING

- Bug tickets
- Feature Requests
- Reporting
  - We can show this to regulators!

REVIEW DAY 2

QUESTION OR CLARIFICATIONS?

PREVIEW DAY 3

SEE YOU NEXT TIME!