

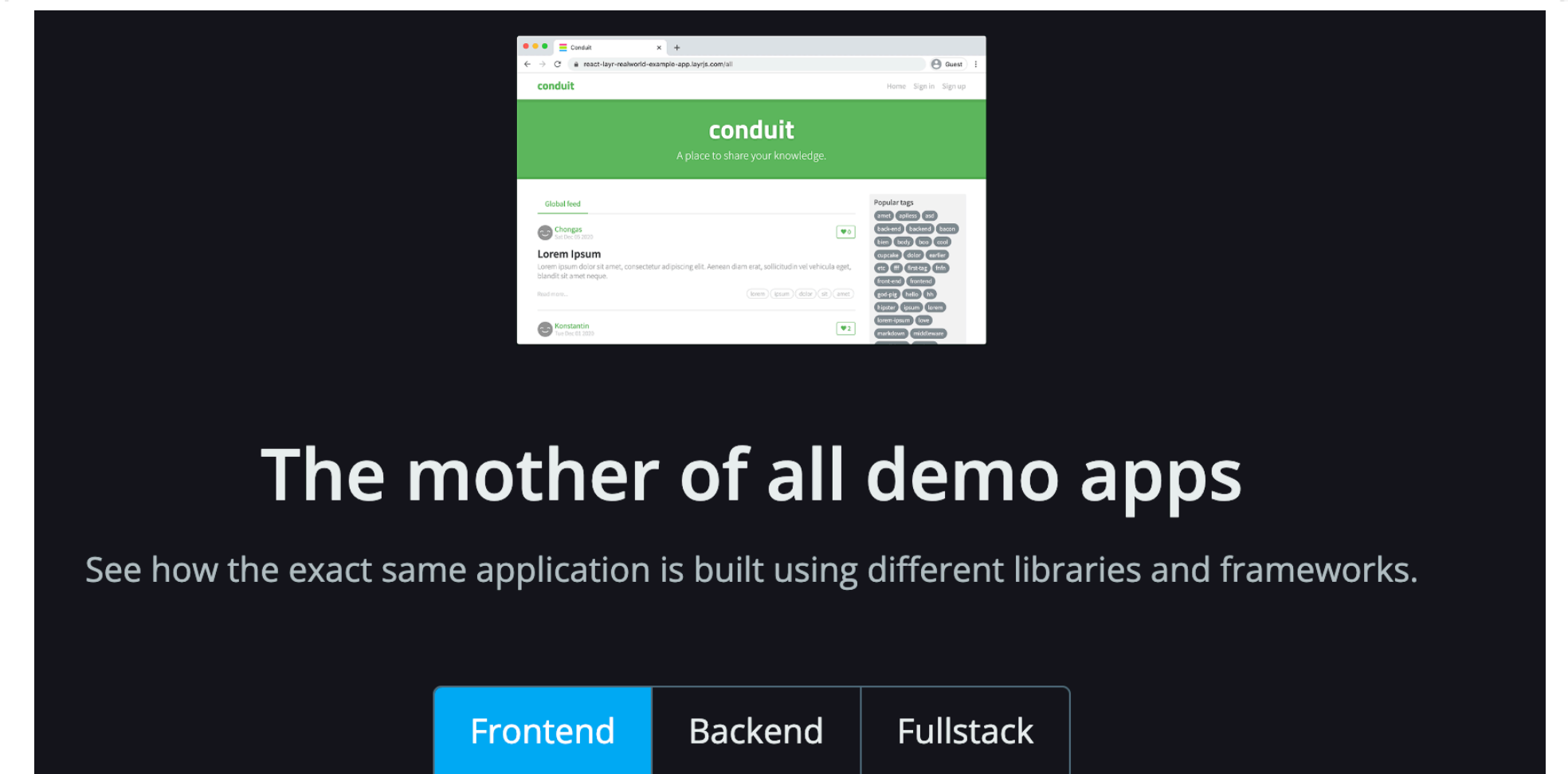
Assignment 1 Specification



Concept, detail and grading
spectrum for Assignment 1












Assignment Motivation & Philosophy

- Assignments present multiple opportunities:
 - Crystallise knowledge on relevant module(s)
 - Opportunity for creativity & innovation
 - Potentially important portfolio contribution
- Its not over when submitted, presented and graded
 - Ongoing maintenance & enhancements
 - Periodically rebuild using new technology stack



<https://github.com/gothinkster/realworld>
<https://codebase.show/projects/realworld>

Assignment 1 Topics

01: Node  The fundamentals of Node. A first Node application.	02: Hapi.js  The Hapi application Framework	03: Joi  Validation with Joi	04: TDD  Test Driven Development	05: Models  Incorporate Mongo Models	06: Assignment 1  Concept, detail and grading rubric
07: APIs  Expose an API from an application	08: REST  Resful API implementations	09: OpenAPI  Documentating OpenAPI/Swagger APIs	10 JWT  Authenticating APIs with JSON Web Tokens	11: Seeding  Seeding the Database	

week no.

1 2 3 rd 4 5 rd 6

Full Stack 1

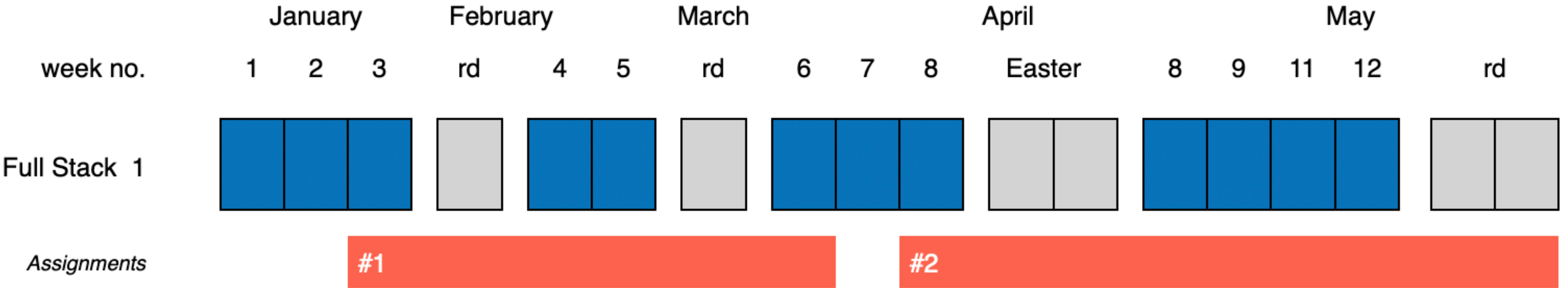


Assignments

#1

3

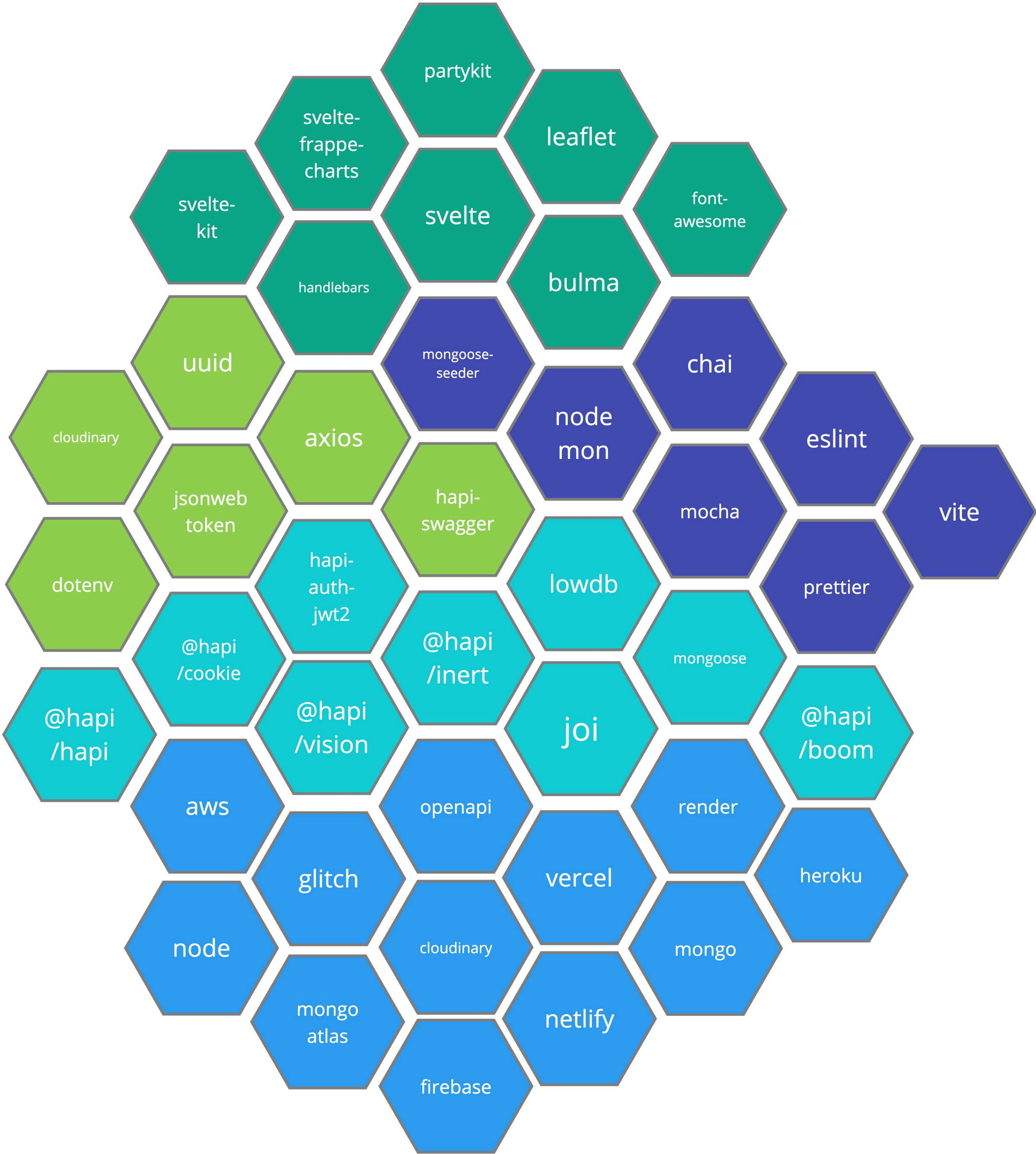
Assignment 1 due March 10th



Assignment 1: March 10
40%

Assignment 2: May 19th
60%

Full Stack Components & Services



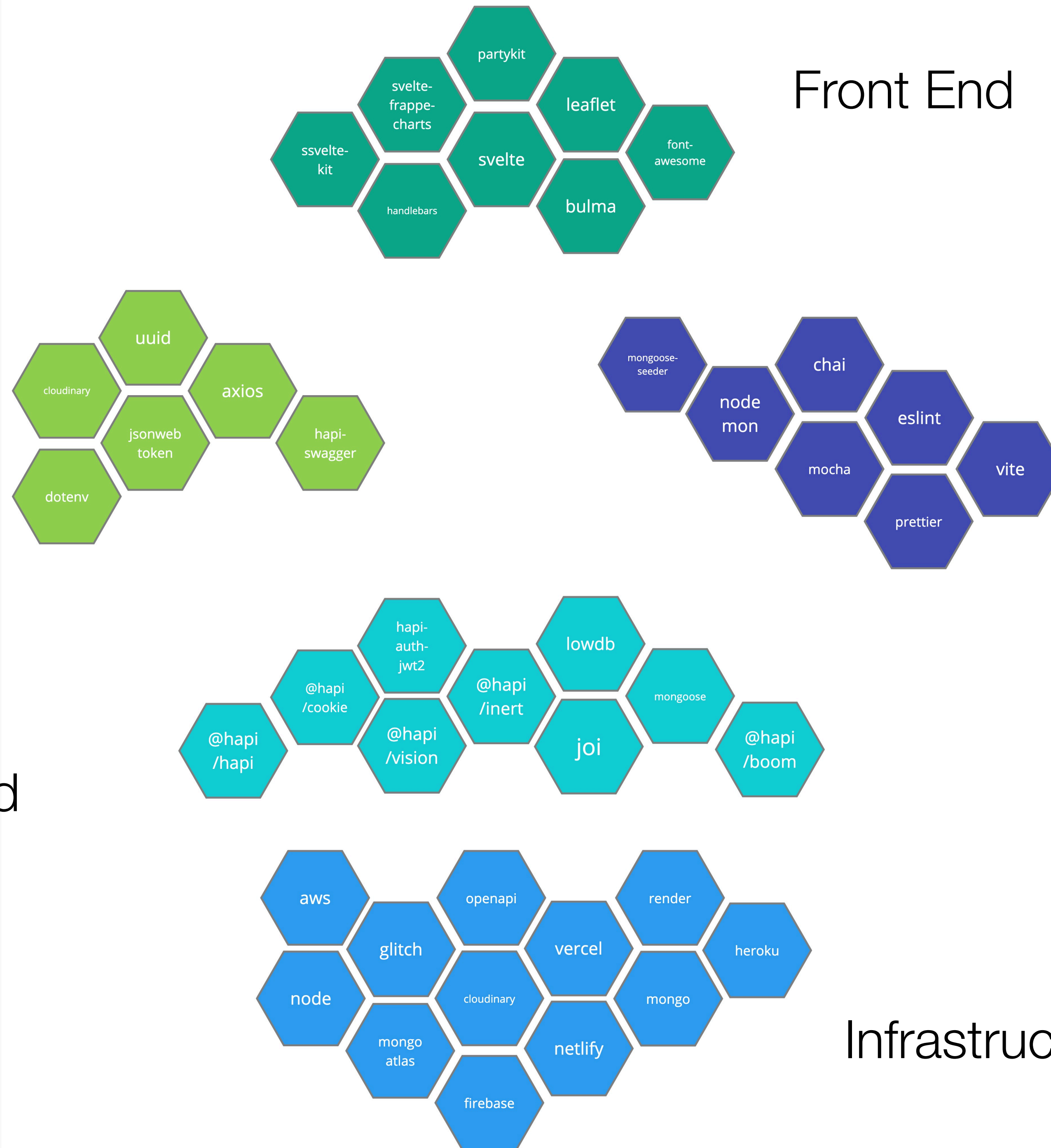
Components

Front End

Tools

Back End

Infrastructure



PlaceMark

A point of interest, or POI, is a specific point location that someone may find useful or interesting.



“A “point of interest” (POI) is a location for which information is available. A POI can be as simple as a set of coordinates, a name, and a unique identifier, or more complex such as a three-dimensional model of a building with names in multiple languages, information about opening and closing hours, and a civic address. POI data has many applications, including augmented reality browsers, location-based social networking games, geocaching, mapping and navigation systems.”

<http://www.opengeospatial.org/projects/groups/poiswg>

https://en.wikipedia.org/wiki/Point_of_interest

https://en.wikipedia.org/wiki/Point_of_interest

PlaceMark Examples

- Landscape feature
- National monument
- Walking Trail
- Bridge
- Tree
- Venue
- Ringfort
- Dolmen
- River
- Bog
- Island
- Forest
- Beach
- Museum
- Climb
- Cycleway
- Birdwatching Locations
- Diving

Product Feature Sets

POI Sharing /
Community Features

Community

- You are free to innovate an the type, range and variety of features

Richer Media
Experience + User
Management

Enhanced

- These categories are to stimulate your brainstorming

Essential Data
Structure & Support
Service

Core

- Some ideas on the next slides...

Information Model

- Placemark
 - Name
 - Category
 - Description
 - Analytics
- User
 - Basic user details

Features

- Sign up / log in to service
- Create, Read, Update, Delete
- Query
- Group into Categories

Core

Information Model

- Placemark
 - Images
 - Location
 - Weather
- User
 - Types

Features

- Placemark images
- Placemark Map Location
- Admin
 - Admin user
 - Admin dashboard

Enhanced

Information Model

- Placemark
 - Creator / Editor
 - Rating
 - Reviews
 - Discussion
- User
 - Interaction information

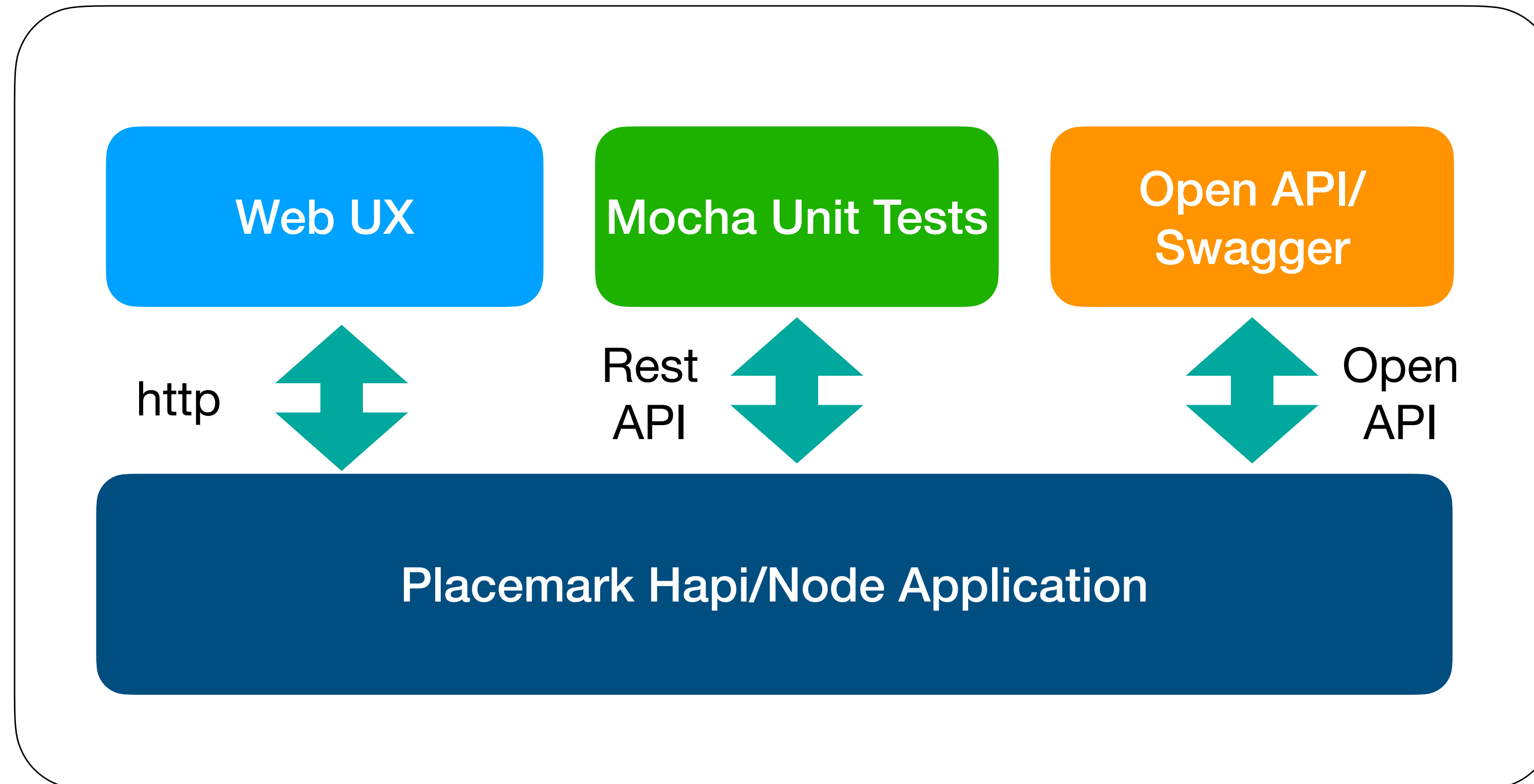
Features

- Share
- Review & Rate
- Notice Board
- Live Updates
- Analytics

Community

PlaceMark 1.0 Architecture

(Assignment 1)



Placemark 1.0 Feature Set

Enhanced Core

Implement POI Core + selected features of POI Enhanced

Support the following User Management Features:

- Sign up / Login in / Delete Account
- User Admin Dashboard with simple analytics

Support the following POI Characteristics:

- Name, Description, Category, Location, Image

Support the following app features:

- Create, Read, Update & Delete POIs
- Organise PlaceMarks into categories

API

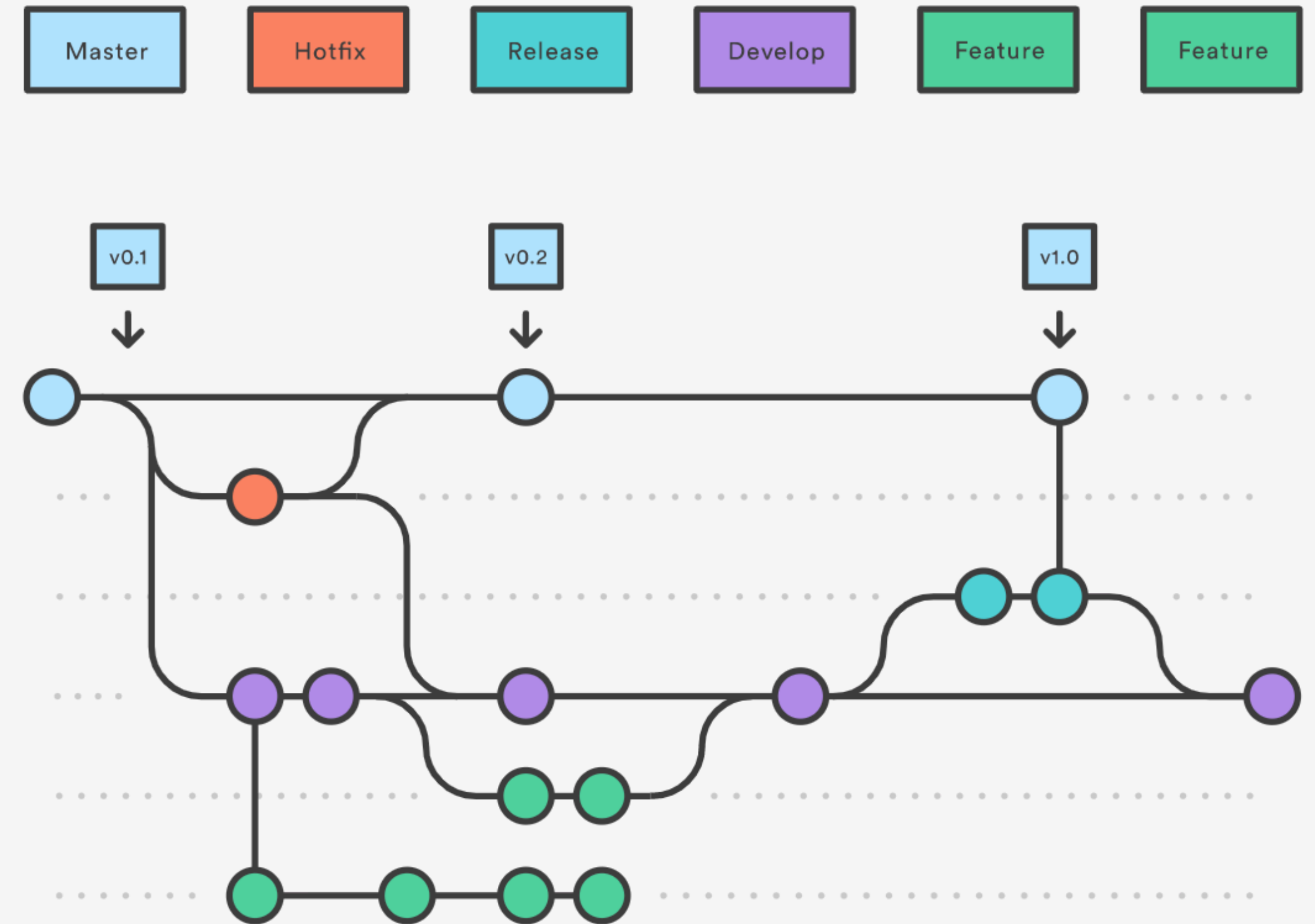
- Endpoints + tests
- Swagger Documentation
- JWT Security

	<i>Accounts</i>	<i>Placemark Features</i>	<i>API/Tests</i>	<i>Models</i>	<i>Deployment</i>	<i>Git</i>
<i>Level 1</i>	Signup / Login	Name	Basic API	Mem	Locahost	git + commit History
<i>Level 2</i>	Cookie Authentication/ User Settings	Description, Location (lat,lng)	Core unit Tests	JSON	Glitch	Structured readme
<i>Level 3</i>	Basic Admin Account List/Remove Users	Categories	Open API (Swagger) + tests	Mongo	Cloud Atlas	Tagged releases
<i>Level 4</i>	Admin Dashboard + analytics	Images	JWT + tests	Firebase	Heroku / Render	Development / Feature / Master branch

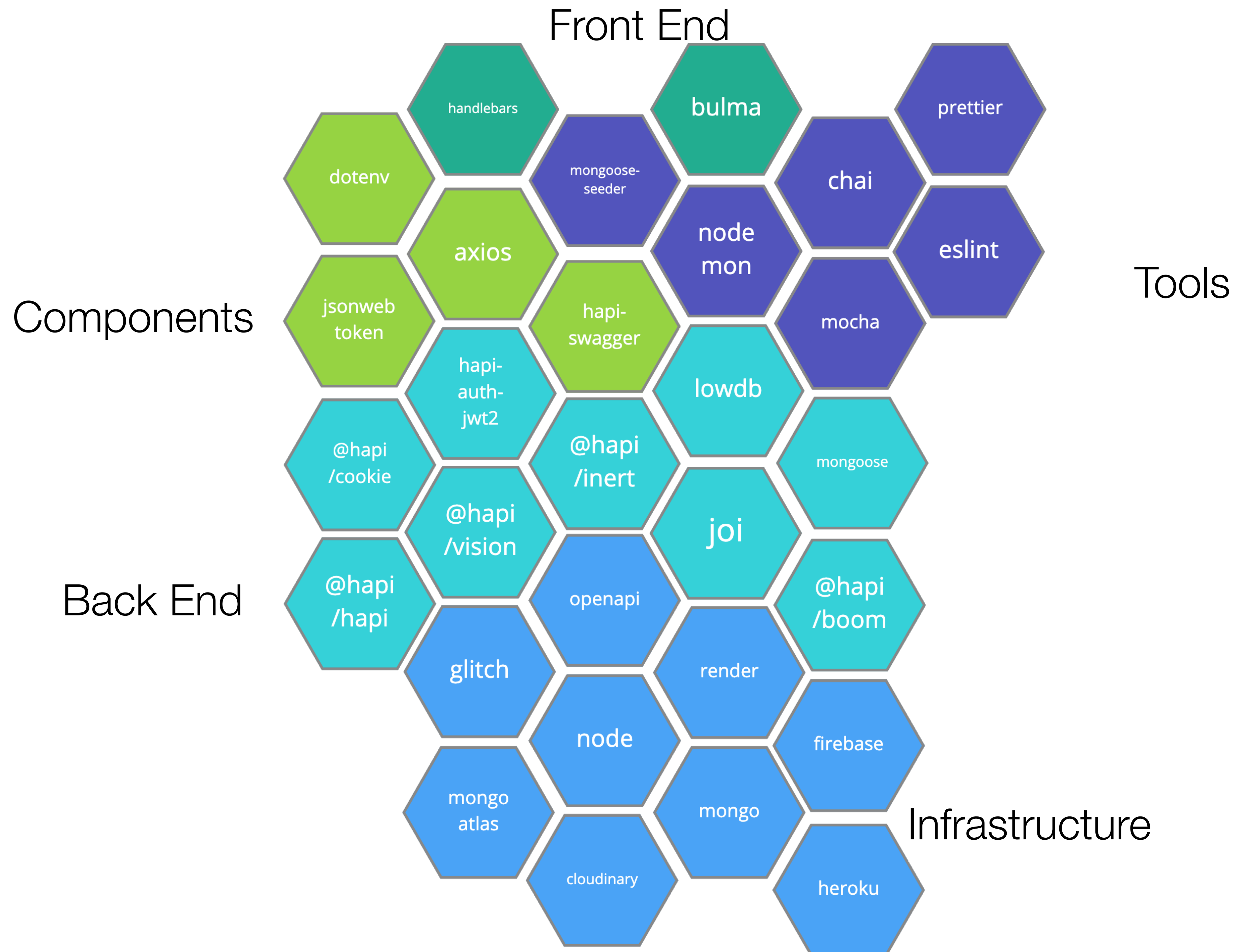
Gitflow Workflow

Gitflow Workflow is a Git workflow design that was first published and made popular by [Vincent Driessen at nvie](#). The Gitflow Workflow defines a strict branching model designed around the project release. This provides a robust framework for managing larger projects.












Gitflow is ideally suited for projects that have a scheduled release cycle. This workflow doesn't add any new concepts or commands beyond what's required for the [Feature Branch Workflow](#). Instead, it assigns very specific roles to different branches and defines how and when they should interact. In addition to feature branches, it uses individual branches for preparing, maintaining, and recording releases. Of course, you also get to leverage all the benefits of the Feature Branch Workflow: pull requests, isolated experiments, and more efficient collaboration.



<https://www.atlassian.com/git/tutorials/comparing-workflows/gitflow-workflow>



Assignment 1 Topics

01: Node  The fundamentals of Node. A first Node application.	02: Hapi.js  The Hapi application Framework	03: Joi  Validation with Joi	04: TDD  Test Driven Development	05: Models  Incorporate Mongo Models	06: Assignment 1  Concept, detail and grading rubric
07: APIs  Expose an API from an application	08: REST  Resful API implementations	09: OpenAPI  Documentating OpenAPI/Swagger APIs	10 JWT  Authenticating APIs with JSON Web Tokens	11: Seeding  Seeding the Database	

week no.

1 2 3 rd 4 5 rd 6

Full Stack 1



Assignments

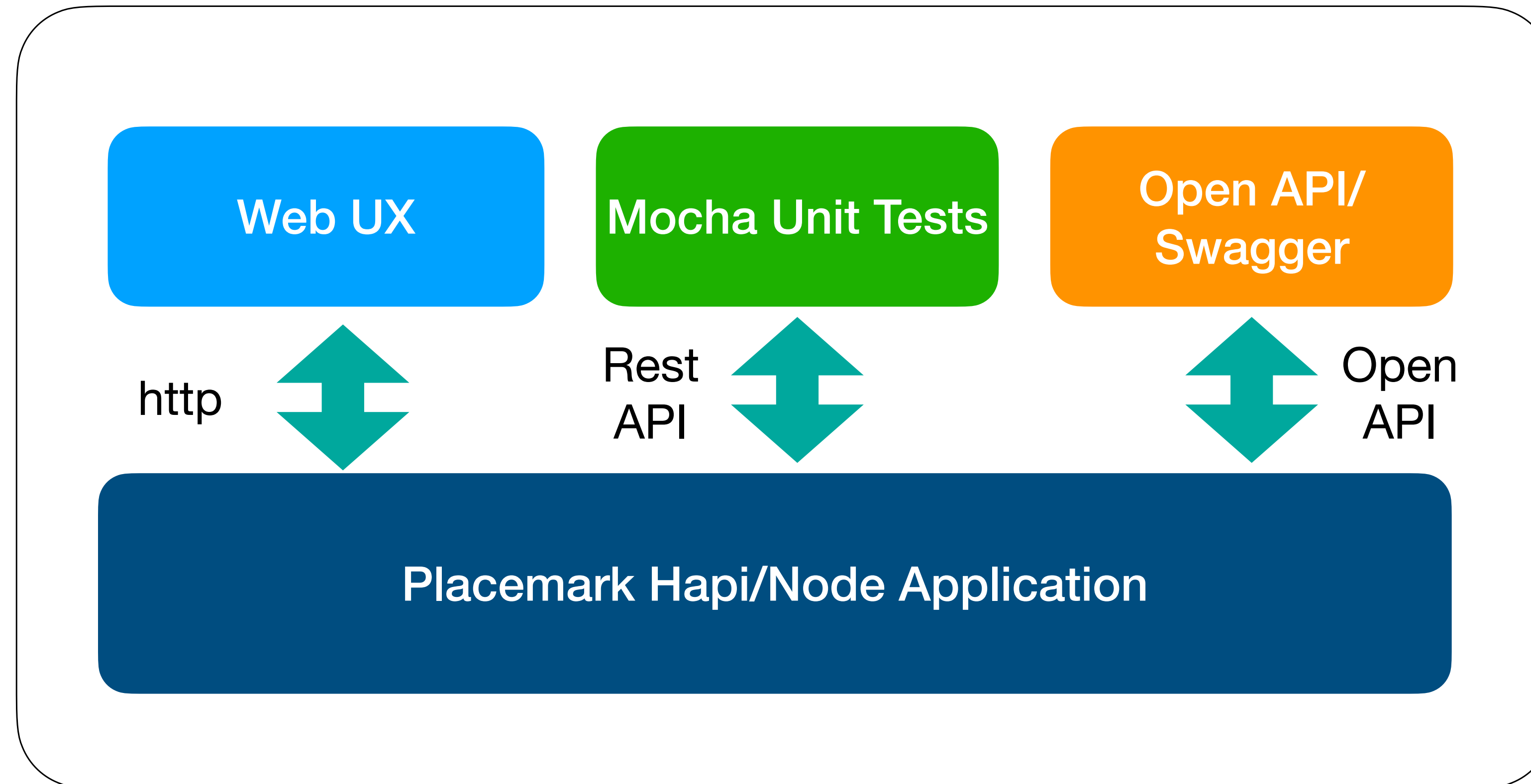
#1

Assignment 1 due March 10th

PlaceMark 1.0 Architecture

(Assignment 1)

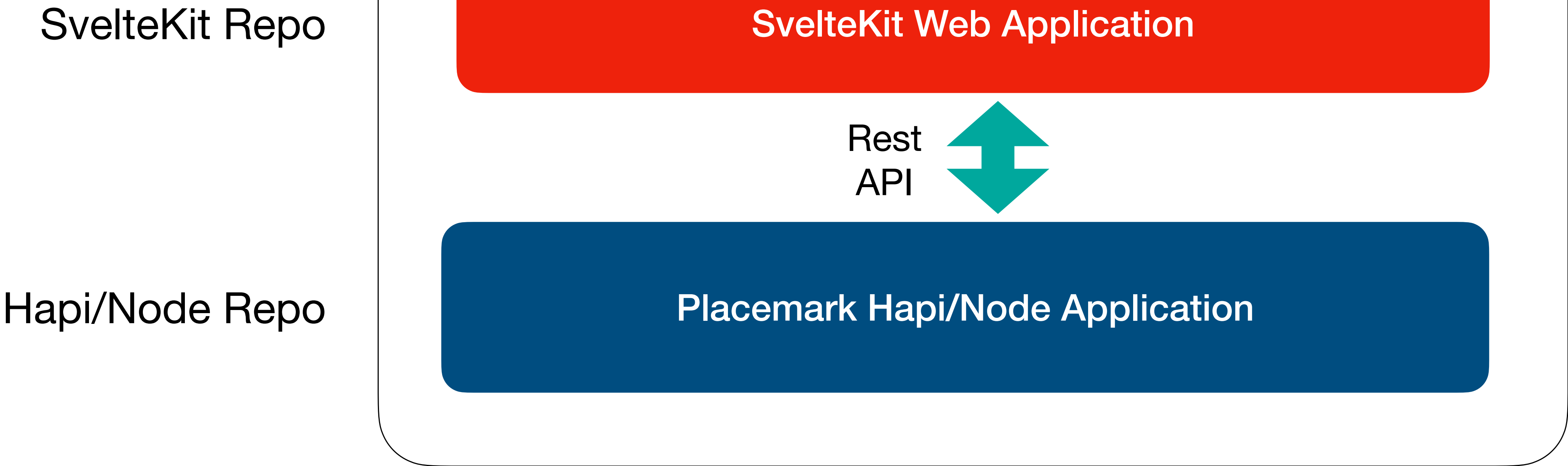
Submitted as a
single repo



	January			February		March			April		May					
week no.	1	2	3	rd	4	5	rd	6	7	8	Easter	8	9	11	12	rd
Full Stack 1																
Assignments	#1							#2								

PlaceMark 2.0 Architecture

(Assignment 2)



	January			February		March			April		May						
week no.	1	2	3	rd	4	5	rd	6	7	8	Easter	8	9	11	12	rd	
Full Stack 1																	
Assignments	#1							#2									

Assignment 1 Specification



Concept, detail and grading
spectrum for Assignment 1