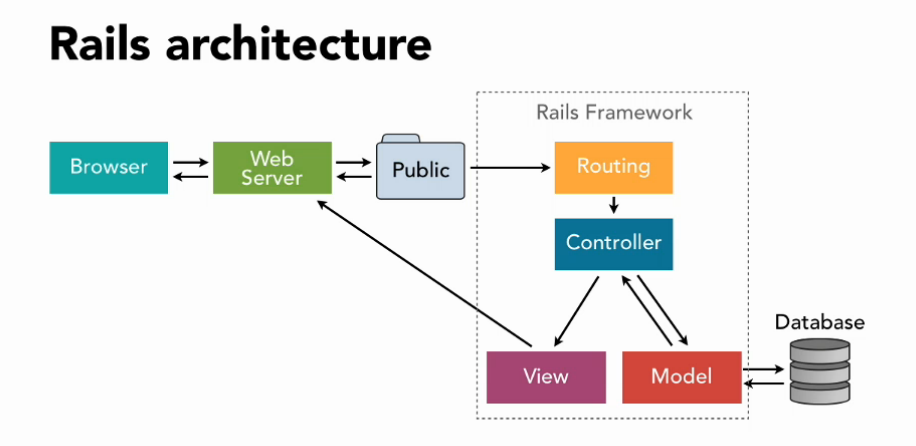
T2A1-A Workbook

Submitted by: Chen Zhang

# Q1 Describe the architecture of a typical Rails application



Ruby on rails is a simple useable framework for full-stack web app development with a database backend. And it follows Model, View and Controller patten which also MVC architecture patten.

* Model – manage the business logic and storage of database. In rails application, model also connect with the database, deal with validation, data storage, data transaction and data table attributions. And it also can add to database, find, update and delete particular data, which we can call it CRUD (create, read, update and destroy).
* View – manage the user interface and representing the information to the customer(end-user). And it is front-end of the application. The view are hybrid files with normal HTML structure, and also use embedded Ruby code.

Assets are normally store and supply visual components such as CSS, JavaScript and other media files (image, wav, movie, etc.)

* Controller – is a connector between View and Model. The requests from browser go through routes, and we use controller to specify the requests (read, create, update or destroy), to call the MODEL for the data then transfer to the VIEW then to the web server.
* Routes – convert URL path into form which more understandable when we develop applications.

To understand the architecture, here s an example, when we input localhost:3000 in browser, it gives the serve (Get) request which go through Routes (config/routes.rb). The it will send the request to the Controller (hello\_controller.rb) and execute the action (index). In action, if it has model action (CRUD) it will find the database then jump to the VIEWS (index.html.erb). then back to the web server.

# Q2 Identify a database management system (DBMS) commonly used in web applications (including Rails) and discuss the pros and cons of this database

Database management system (DBMS) normally use to add, access and process data in computer database.

MySQL is one of most popular DBMS helps developers with their applications. And there are advantages and disadvantages of MySQL.

### Advantages

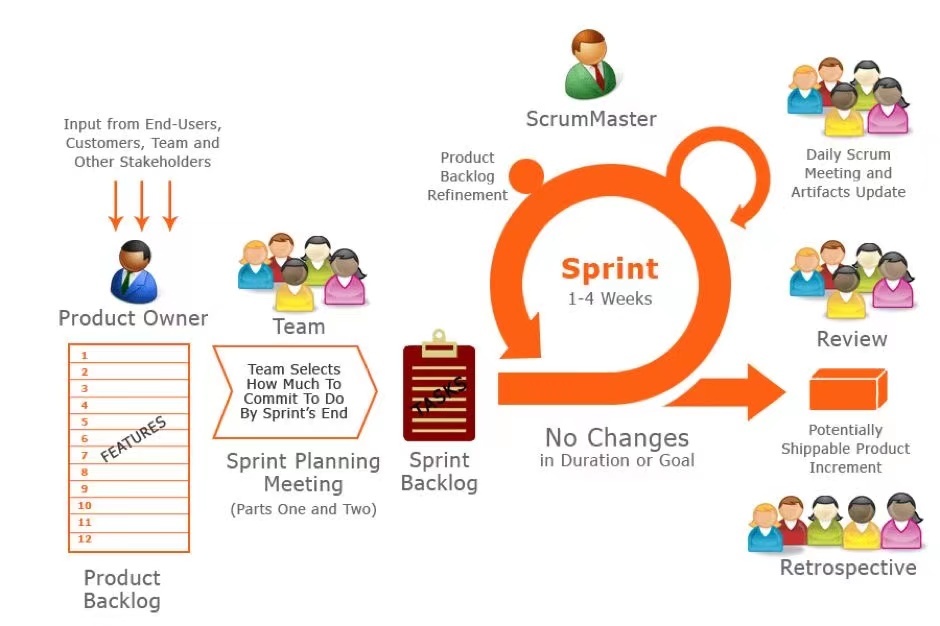
* Small volume, quick speed, low cost of ownership, open source, support for multiple operating systems.
* Is open-source database, the interface to support multiple language connection operation.
* The core thread it uses is completely multi-threaded and it support multi-processor. And it threads are lightweight processes that can flexibly serve users without excessive system resources.
* a very rapid and stable memory allocation system based on thread, can keep using it without consider about the stability.
* MySQL has a very flexible and secure permission and password system. When a client connects to a MySQL server, all password transfers between them are encrypted, and MySQL supports host authentication.

### Disadvantages

* Hot backup not supported.
* The biggest drawback of MySQL is its security system, the main reason is it complex and non-standard, and only can be changed when you call mysql admin to re-read the user rights.
* If you have a large number of stored procedures, the memory that used for each connection to these stored procedures will increase significantly. In addition, if you overuse a large number of logical operations in a stored procedure, CPU utilization increases. This is why it is not efficient for very large database.
* The price of MySQL may vary with different platform and installation. And compare to other paid database management systems, it lacks a good developing and debugging tool.

# Q3 Discuss the implementation of Agile project management methodology

Agile project management is an iterative and incremental solution for development which aims to deliver value by seeking feedbacks and changes by customers and continuing improving the project.

Scrum is the most popular Agile project management according to Agile Report. 

The Scrum framework is composed of three characters, five events and three artifacts and describes how these entities to connect with each other and implementation framework.

Three characters

Scrum masters - plays a critical role in the Scrum team by ensuring that the team uses the right processes, holds the right meetings, and helps everyone understand Scrum theory, practices, rules, and values.

The Product Owner - is the core of authorized Product leadership and is one of the three roles of the Scrum team. Po plays the role as product manager.

Team - all of the people are referred as Engineers and five to seven people is a proper amount, and includes with multiple roles for product, design, front-end, back-end, testing, and so on. Team is the actual value producer;

Three artifacts

Product Backlog – is priority list for products.

Sprint Backlog – current project, only in one sprint cycle.

Increment – is the sum of all Product Backlog items completed during Sprint and the incremental value of all previous sprints.

The Scrum implementation process

1. Before Scrum – make sure the project suit for Scrum.
2. Identify the basic requirements of the iteration – iteration need a fix time period, no more than 6 weeks
3. Prepare Sprint plan – backlog must exist; for one project or product you can only have one backlog and one product owner; all important backlog entries have been scored based on importance.
4. Create Sprint plan – Sprint planning meeting is the key and most important events in Scrum, this meeting will achieve:

Target of Sprint; Team members; Sprint backlog; Determine the Sprint presentation time; Daily Scrum time and target; decide which features to complete during the sprint.

1. Sprint planning meeting - Set sprint goals and a backlog for the given product
2. Daily Scrum meeting – focus on Sprint target, normally with three questions:

What have I done to help achieving the target yesterday?

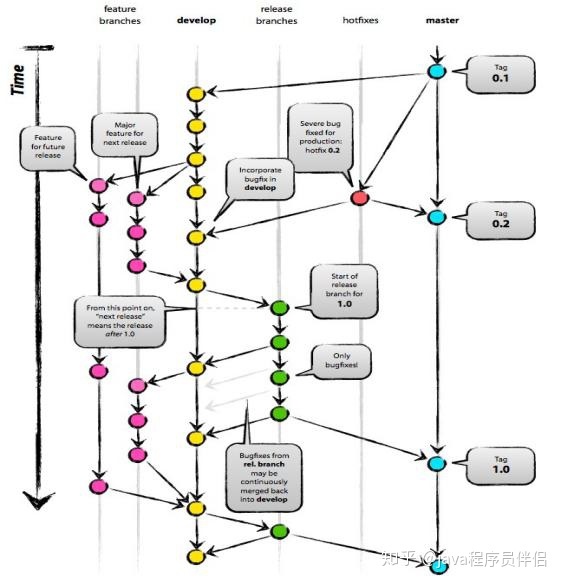
What will I do for today?

Anything slow me or team down to achieving the Spring target?

1. Review – Teams demonstrate the Increment during the Spring.
2. Spring review meeting – figure out where did go good and where did go wrong from last spring.

And how to improve it for next Spring.

# Q4 Provide an overview and description of a standard source control workflow

Source control helps developer to control and manage changes code or versions in their program and make sure that they are on the right version of the source code.

A standard process for managing source control is as above picture.

Support branches:

The feature branch - is used to develop features and must be pulled from the latest develop branch code. The branch naming is basically feature.

Release branch - When the develop branch already has all the code on the line this time, and when all the tests have passed, a release branch can be created from the develop branch. The release branch is designed for releasing new product versions.

Hotfix branch - When there is a bug online that needs to be fixed urgently, the hotfix branch is derived from the current master branch.

Main branches are Master Branch and Develop Branch

Master branch - only stores online code (final code). Only when it is confirmed to go online, then merge it into the master branch and tag it (version 1.0, 2.0).

Develop branch - is the final branch at the time of development and has all the features that need to go live for all current versions. When develop team have developed different requirements in their own feature branch and need to merge tests, then we need to merger all feature branches into Develop Branch. Then submit the test, and fix the bug on the develop together. Then push the Develop Branch to Release Branches.

Source Control is keeping the codebase clean and tidy and also isolating the development among the members to avoid inefficiency and confusion caused by the mutual influence of the code in development.

# Q5 Provide an overview and description of a standard software testing process (e.g., manual testing)

Software testing process is the process of operating or measuring a software system, using manual or automatic ways, to verify that the software system meets specified requirements and to identify deviations from expected results. And also, software testing is the process of judging and evaluating the quality or usability of software using by certain methods like manual testing, black box testing, static and dynamic testing, etc.

The steps in software testing:

1. Refine the test requirements according to project and product requirements.
2. According to the test requirements and the overall plan of the project, formulate the test plans which includes test time arrangement, human resource arraignment, test strategies, and conduct review.
3. According to the test requirements and related design documents, write test cases, specific operation steps of each test point, expected results, and review the use cases.
4. Prepare test environment and test data, including hardware environment and software environment for test system deployment.
5. Execute test cases, submit bugs found during testing, and perform regression testing through version iteration to verify related bugs.
6. After completing the functional test of the internal software system and the system test, submit to the customer for acceptance test
7. Write software test reports
8. Summarize the testing process and file all documentation during the testing process.

# Q6 Discuss and analysis requirements related to information system security and how they relate to the project

# Q7 Discuss common methods of protecting information and data and how you would apply them to the project

# Q8 Research what your legal obligations are in relation to handling user data and how they can be met for the project

# Q9 Describe the structural aspects of the relational database model. Your description should include information about the structure in which data is stored and how relations are represented in that structure.

# Q10 Describe the integrity aspects of the relational database model. Your description should include information about the types of data integrity and how they can be enforced in a relational database.

# Q11 Describe the manipulative aspects of the relational database model. Your description should include information about the ways in which data is manipulated (added, removed, changed, and retrieved) in a relational database.

# Q12 Conduct research into a marketplace website (app) and answer the following parts:

### List and describe the software used by the app.

### Describe the hardware used to host the app.

### Describe the interaction of technologies within the app

### Describe the way data is structured within the app

### Identify entities which must be tracked by the app

### Identify the relationships and associations between the entities you have identified in part (e)

### Design a schema using an Entity Relationship Diagram (ERD) appropriate for the database of this website (assuming a relational database model)