**Royal Pharmaceutical Society(RPS)**

(FCPR)

TABLE OF CONTENTS

[1. Approach for Implementing FCPR 3](#_Toc427943429)

[1.1 High level Architecture 3](#_Toc427943430)

[1.2 Play with Rest Services 4](#_Toc427943431)

[1.3 Data Migration Strategy 4](#_Toc427943432)

[1.4 Development Approach 5](#_Toc427943433)

[1.5 Sprint 0 - Architecture, Design and Requirements 5](#_Toc427943434)

[1.6 Development Sprint - Development and System Testing 7](#_Toc427943435)

[1.7 Stabilization Sprint 7](#_Toc427943436)

[1.8 Final Sprint - User Acceptance Testing (UAT) and UAT preparations 8](#_Toc427943437)

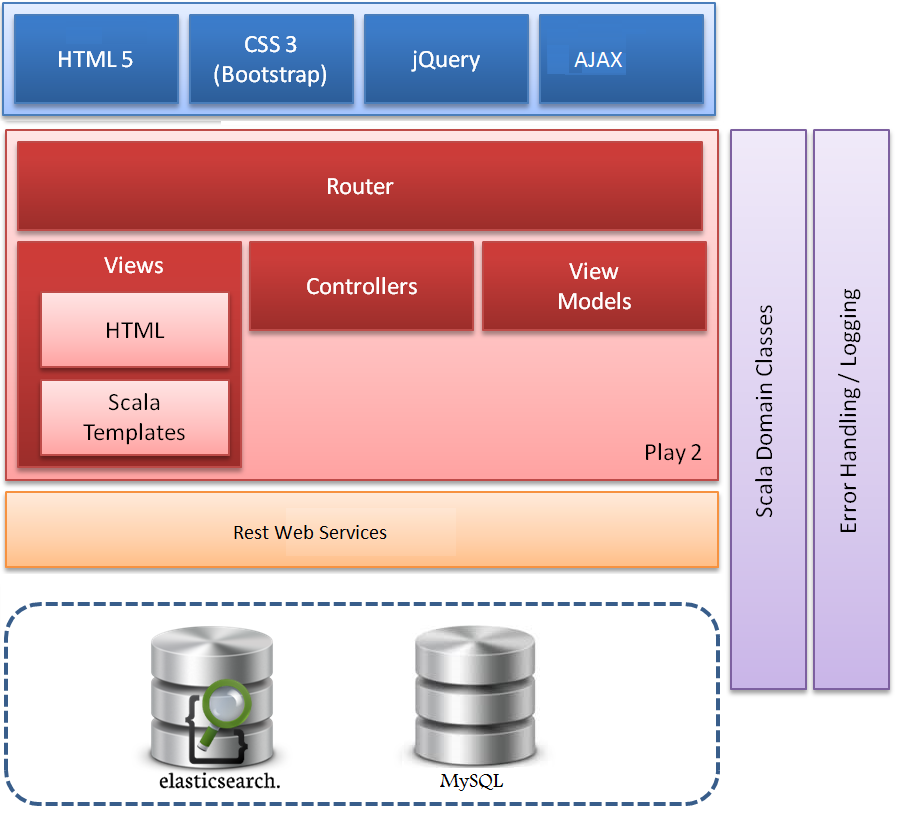
[1.9 Project Communication 8](#_Toc427943438)

[1.10 Version control 10](#_Toc427943439)

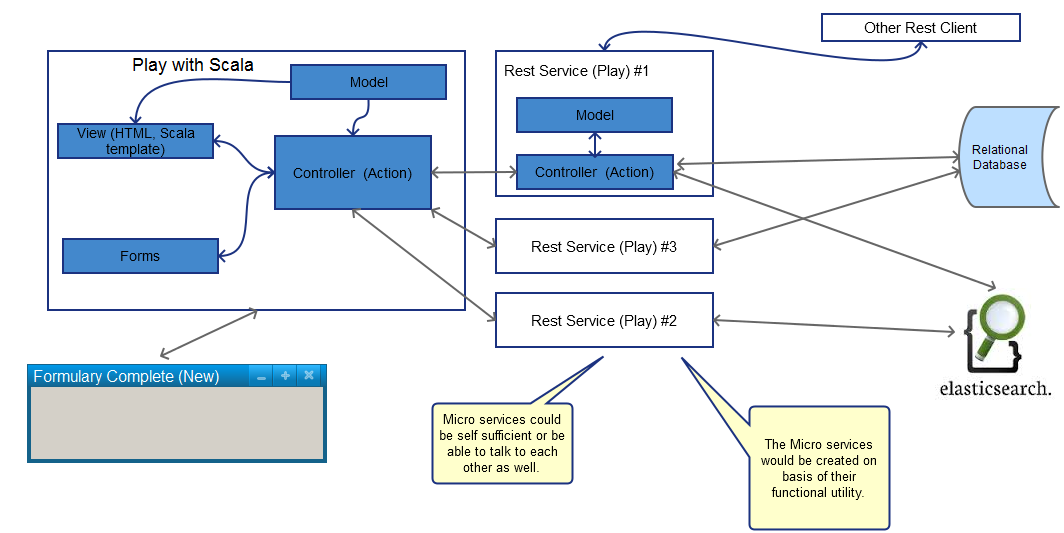
[1.11 Risk/Issues 10](#_Toc427943440)

# Approach for Implementing FCPR

## High level Architecture



## Play with Rest Services

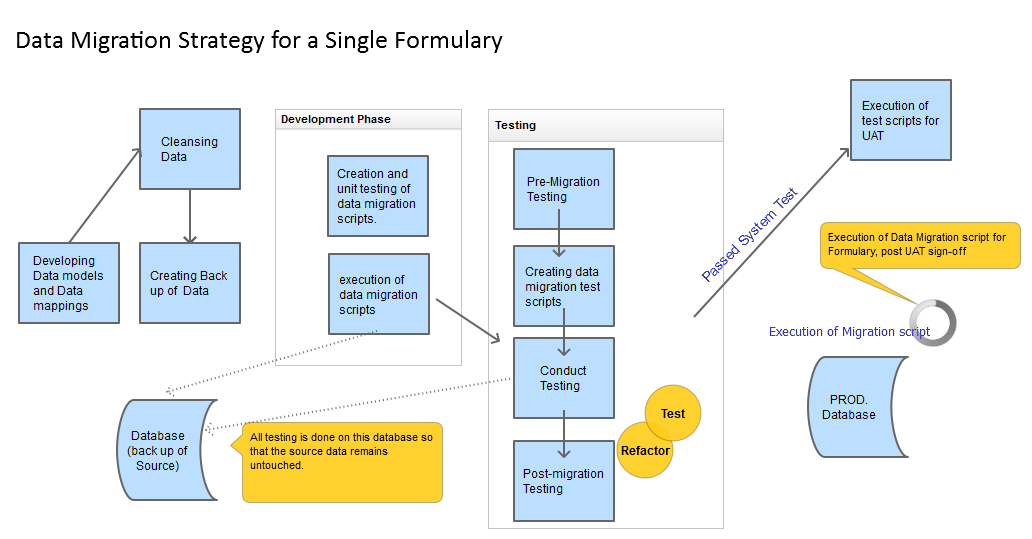


The application will involve use of Rest Services (in Play Framework).

The controller will invoke the Rest Services depending on the requirement at run-time. These Web APIs will access the data (from ElasticSearch and/or any other relational database) and respond back to the client (Application controller).

If required the controller would make multiple calls to the Rest Services, collate information and pass on the response to web front end

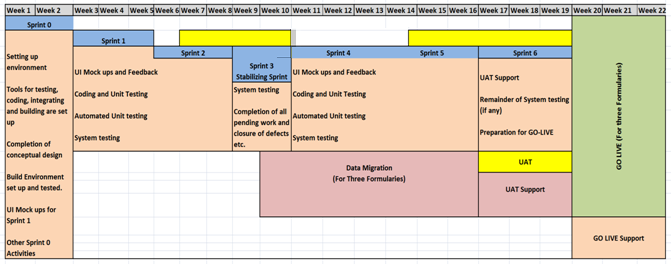
## Data Migration Strategy



## Development Approach

Rave proposes the sprint delivery with the following:

* [Sprint 0](#_Sprint_0_-_1) (2 weeks )
* [Development Sprints](#_Development_Sprint_-) (3 weeks each)
* [Stabilization Sprints](#_Stabilization_Sprint) (2 weeks) –
* [UAT Sprint](#_Final_Sprint_-)  (2 weeks)



UAT would start in middle of sprint 2 (on sprint 1 build), and continue over the stabilization sprint. The next UAT cycle would start in middle of second sprint after completion of each Stabilization Sprint. (in Sprint 4, Sprint 6 and so on)

After UAT Sprint, Rave would provide support for GO-LIVE activities, and also provide POST GO-LIVE support.

## Sprint 0 - Architecture, Design and Requirements

### Key Activities

This Sprint will primarily involve close interaction with RPS’s stakeholders to understand business objectives, critical success factors, business priorities, release plan, along with detailed requirements.

This Sprint 0 will be **conducted on-site** and the technical lead, project manager will be on-site for the full duration of the Sprint.

The Technical Lead will be responsible for producing Architecture Document, DB design and completing POCs (if required). He would also engage with RPS Technical Team to provide confidence that the code developed would be efficient, flexible, and scalable.

The Project Managers will interact with their counterparts to come up with a detailed sprint plan. They would resolve any impediments and mitigate the current and potential risks.

The Product Managers will ensure that all functional and non functional requirements are captured within the Acceptance criteria. The Sprint 0 will be utilized to produce concrete backlog.

The following will be accomplished:

• Discussions with stakeholders and business team to get further insight into functional requirements.

• Identifying dependencies and Risks and defining plan to mitigate those.

• Requirements analysis and creation of design, PBIs (Product Backlog Items for next sprint/s)

• Software architecture based on agreed functional and non-functional requirements (FRs & NFRs)

• Identification of dependencies with all the stakeholders

• Identification of Single-Point-Of-Contact for all the stakeholders

• Software architecture based on agreed functional and non-functional requirements (FRs & NFRs)

* Setting up environment
* Tools for testing, coding, integrating and building are set up.
* Completion of conceptual design, along with POC (if required).
* Build environment set up and tested.

## Development Sprint - Development and System Testing

### Key Activities

This Sprint will primarily involve development and system testing of Acceptance Criteria as captured in the approved user stories .

### Deliverables for this stage

The following would be delivered as output of development sprints

• Sprint Plan

• System tested source code build (for deployment and testing in RPS environment)

• Release Notes document

* System test cases
* Automated unit test cases.

### Expectations from RPS

During Development Sprints, Rave would expect the following from RPS :

• Sharing review comments during the periodic showcases

• Review and approval of PBIs for next Sprint.

• Testing of interim builds and providing feedback for facilitating early actions from Rave

**Notes:**

1. All Development sprints will be progressive in nature and will involve similar activities, deliverables and expectations as detailed above.
2. During Sprint 2, RPS is expected to carry out interim UAT on Sprint 1 build and log issues in the defect tracking tool. Similarly during Sprint 3, RPS will carry out interim testing on Sprint 2 build and so on.
3. Post Sprint 2, Rave proposes two weeks of stabilization sprints.

## Stabilization Sprint

After running every two consecutive full development sprints (#1 and #2), Rave proposes a two week long stabilization sprint.

**Objective:**

Rave to fix high priority issues (defects or suggestions) reported during previous development sprint’s UAT cycle.

Complete development and system testing (if pending from earlier sprints)

Closure of defects.

**Benefits:**

1. Further stabilize the working software developed so far.
2. Further reduce the load at the time of final UAT cycle during UAT Sprint
3. Reduce rework by not accumulating too much pending work, if any.
4. Provide a realistic progress measure against planned project progress.

**Deliverable:**

Updated build for interim UAT

## Final Sprint - User Acceptance Testing (UAT) and UAT preparations

### Key Activities

This Sprint will primarily involve the final round of acceptance testing by RPS’s team on-shore for a period of four weeks. Issues reported during acceptance testing will be fixed by the off-shore development team and re-tested. Final preparation for Go Live is planned, post completion of acceptance testing.

The following will be accomplished:

• Acceptance testing by RPS team

• Support from off-shore by Rave’s development and testing team

• Rave will support testing by resolving bugs, if any, and providing clarifications.

### Deliverables for the phase

• Code Deployment Package for Production Releases

• Release Notes document

• Final updated technical design document (Architecture, Object model, Data model).

### Expectation from RPS

• RPS’s relevant stakeholders to be made available for the deployment of the build in their environment

• RPS will test the deliverables in the applicable environment and report issues / defects within the issue tracking tool.

• Final acceptance of deliverables and project sign-off.

## Project Communication

The Rave team will interact with The RPS team on an as needed basis for resolution of issues/clarifications, if any.

**Overlapping Working hours:** Working hours of the off-shore team will be adjusted suitably during the critical phases of the project to facilitate sufficient overlap (of 3-4 hours at least) with the normal business hours of RPS.

**FTP Server:** All the deliverables will be uploaded to the FTP server hosted at Rave. The will be provided with a login account to access the FTP server.

**Status Reporting:** Status reports showing the progress, risks, issues, and concerns will be sent out to RPS

**Chat:** Rave’s project team, if required, may communicate with the authorised person (s) at through Skype/Google Hangout or any other instant messaging client, for speedier resolution of issues and clarifications.

**Video Conferencing:** Video Conferencing would be used to review project artefacts and application demonstration. Rave’s suggested approach to video conferencing is a combination of conference bridge for voice calls and Webex for screen sharing and videoProject Reporting Mechanism

JIRA has been fixed as tool for maintaining the Product and Sprint backlog, tracking Sprint progress, reporting on team progress and Defect Management.

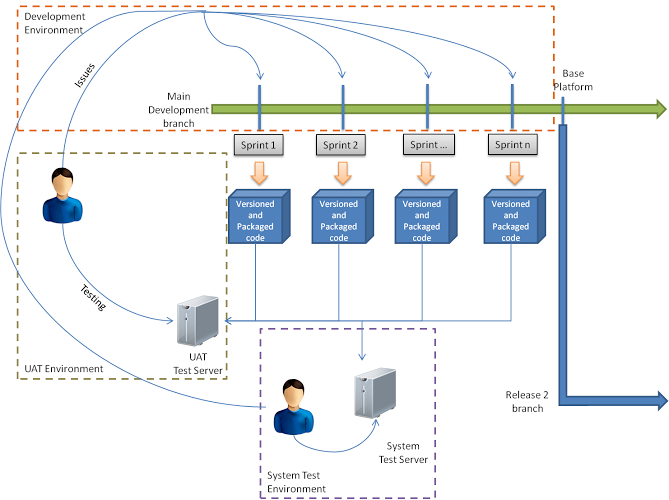
All stakeholders would have access to the tool being employed and will be able to track progress at any given point of time. The scrum master will ensures that the team members update their efforts against each Sprint Backlog Task, so that the reports on progress are accurate. The scrum boards are also maintained.

The daily stand-ups are conducted and each team member gives details relating to progress of tasks in hand along with impediments, if any**.** Based on the inputs obtained from the team, the scrum master highlights any concerns, with respect to sprint delivery, as early as possible to the stake holders.

Show and Tell sessions are conducted so that stake holders have a clear view of the status of the tasks and functionality being delivered. These sessions are also done during the Sprints, which give visibility and more time to take corrective actions within the Sprint duration itself.

Pre-Planning will be done in middle of the last week of sprint, so that stake holders are aware of the roll-over (if any) from current sprint and the tasks likely to be delivered in next sprint. This woudl help the stake holders get a general view of percentage of work completed with respect to the project goals, and also understand the remedial measures taken, if the project is off-track.

## Version control

i

As depicted in diagram above, code is checked in Main Development branch, and deployed for system testing. The code is labelled at end of each sprint and deployed to UAT.

The issues identified during system testing and UAT are bought back to the development teams as defects/user stories. These are fixed and released in the next deployment to system test/UAT.

Rave envisages that there will be only one live release at the end of Base Platform development, at that point in time a different branch of the code will be defined and baseline-d and any code fixes as reported by QA or Service desk can be run (if required) on that code branch.

## Risk/Issues

| **Risk** | **Mitigation** | **Contingency** |
| --- | --- | --- |
| JSon structural Changes | The structure for JSon should be finalized by end of Sprint #0. | Any changes later will have to be taken up as Change Request, as per the impact of the changes. |
| Support for IE 7 | Give a message to the user that IE 7 are not fully supported. |  |
| Timely availability of RPS members for Review, Acceptance and Sign-off. | The RPS to ensure the availability and commitment from each member. Schedules will be planned in advance so that members can plan their work accordingly. | Impact on schedule |
| User Story cards not approved before start of development . | Story cards will be prepared and approved one sprint ahead of its development. | De-prioritise the feature and scope it for subsequent sprints. |
| Show stopper issue reported during interim UAT, at - RPS, that puts further testing on hold. | Rave’s development team will address the issue ASAP and send the patch for deployment and testing |  |
| Schedule slippage due to factors beyond Rave’s control | Proactive project tracking and reporting by Rave to RPS. | Extend the project beyond agreed end date. |
| Change (addition/ modification) in signed-off requirements | Mutually discuss the impact and decide if any requirements could be de-prioritised. | Cost and schedule impact. If same would require any additional efforts then these would be considered as Change Request (CR) and appropriate change control procedure will be applicable. |