

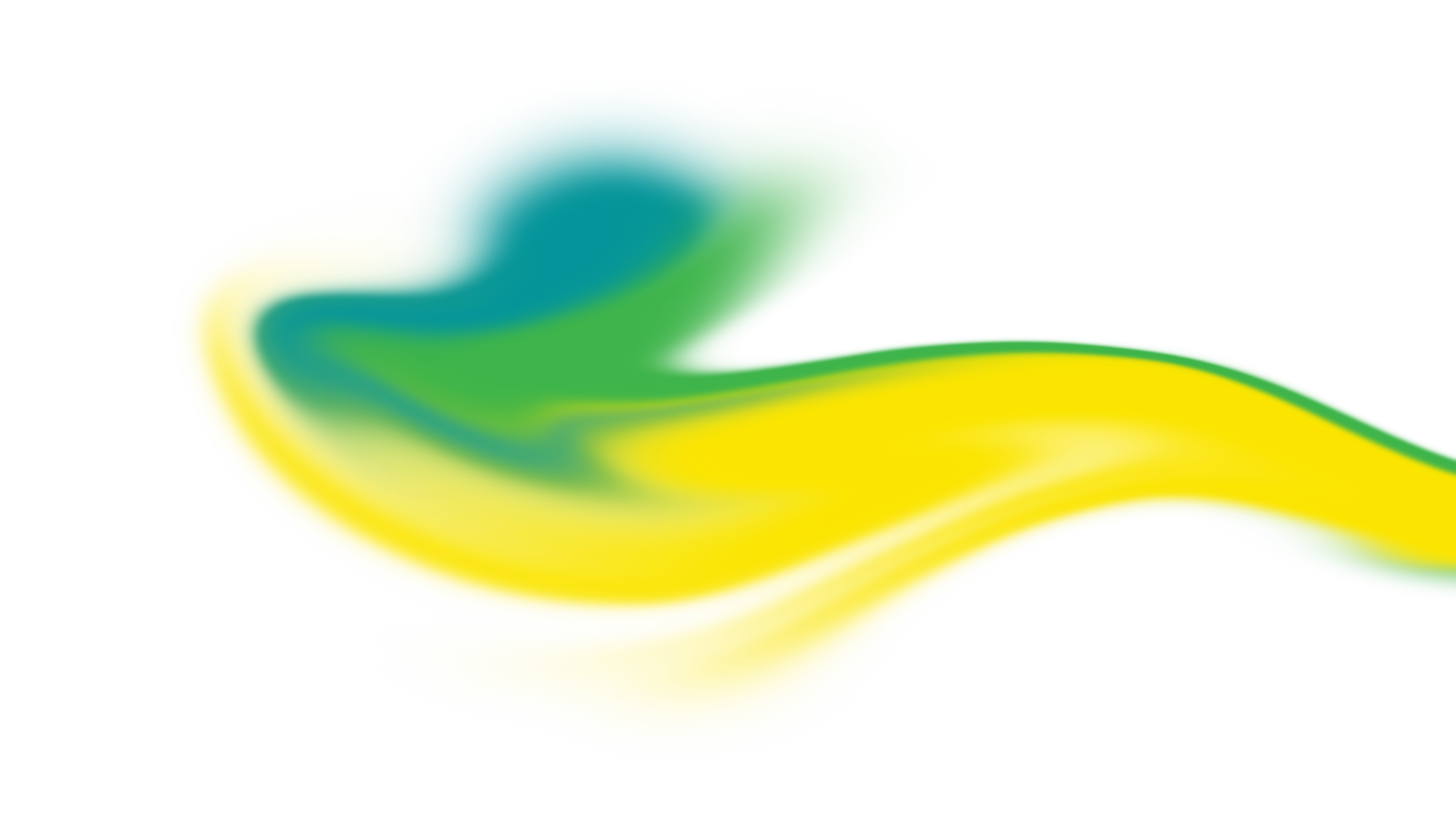
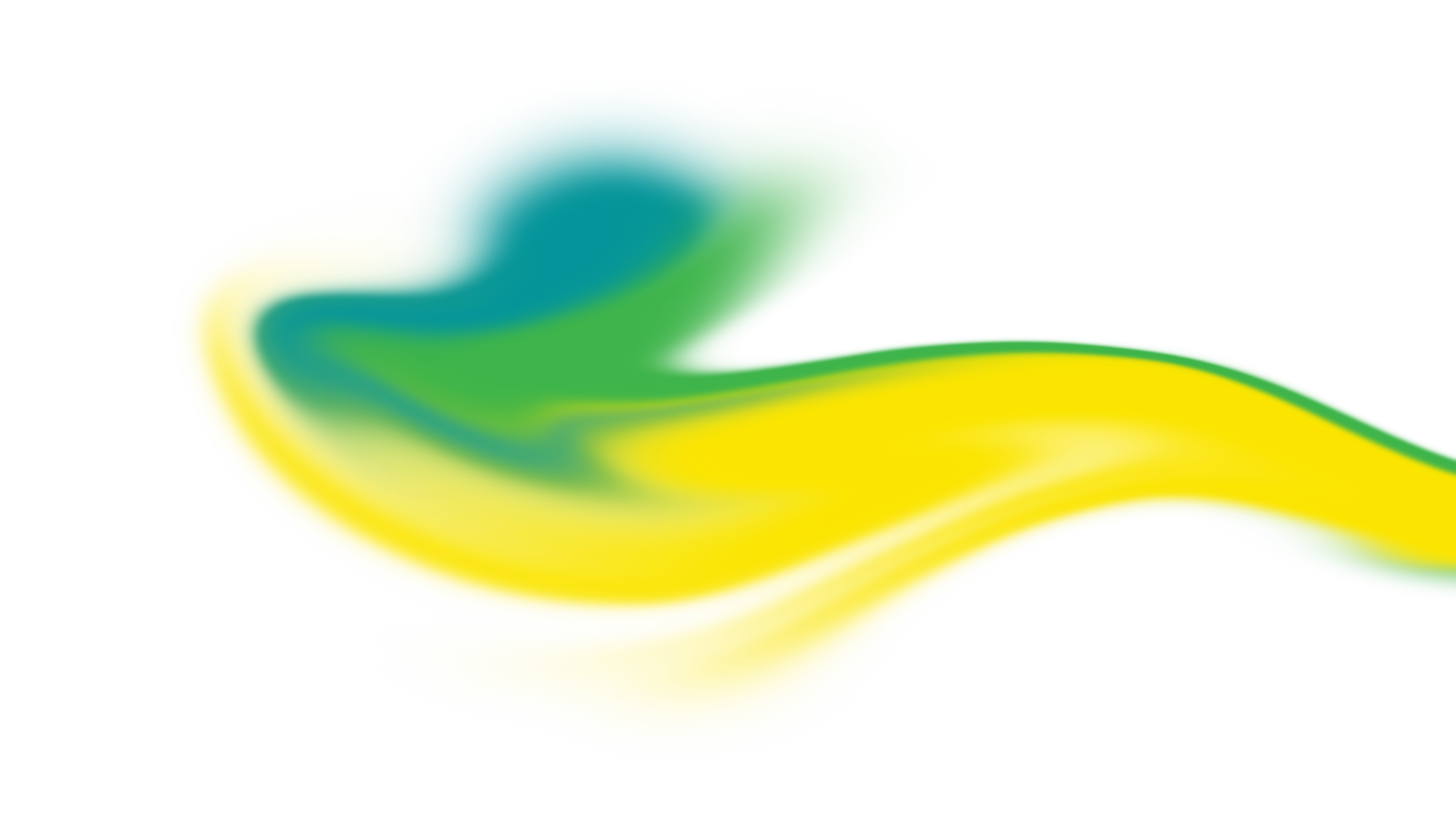
Developing Viatris

Presentation Manager and Viatris Email Agent

**Submitted by**

**Happiest Minds**





**Copyright Information**

This document is the exclusive property of Happiest Minds Technologies Ltd. (“Happiest Minds”). The recipient agrees that they may not copy, transmit, use, or disclose the confidential and proprietary information in this document by any means without the expressed and written consent of Happiest Minds. By accepting a copy, the recipient agrees to adhere to these conditions to the confidentiality of Happiest Minds practices and procedures.

**Confidentiality Clause**

This document is being submitted to Viatris Inc. (“Viatris”) by Happiest Minds Technologies Ltd. on the understanding that the contents of this document will not be divulged to any third party without the express written consent of the parties. It is also understood that the parties will not divulge any confidential information about Viatris that it may have access to during this interaction.

**Disclaimer**

This document has been prepared based on the information provided by Viatris. Wherever proposed, the solutions and/or services mentioned are based on the requirements defined and understood by us at the time of preparing this document. While every effort has been made to make this document as accurate as possible, there might be changes to the document based on the subsequent discussions.

Table of Contents

[1. Requirement Details 4](#_Toc128655333)

[1.1. Our Understanding of Requirements 4](#_Toc128655334)

[1.2. In Scope 6](#_Toc128655335)

[1.3. Out of Scope 6](#_Toc128655336)

[1.4. Supported OS and Browser 6](#_Toc128655337)

[1.5. Assumptions 6](#_Toc128655338)

[1.6. Dependencies 7](#_Toc128655339)

[2. Proposed Solution 7](#_Toc128655340)

[2.1. Viatris Email Agent Architecture 7](#_Toc128655341)

[2.2. Presentation manager architecture 9](#_Toc128655342)

[2.3. Testing Approach 11](#_Toc128655343)

[2.4. DevOps Approach 13](#_Toc128655344)

[2.5. Coding measures 13](#_Toc128655345)

[2.6. Technology Stack – Indicative 13](#_Toc128655346)

[3. Project Delivery Approach 13](#_Toc128655347)

[3.1. Governance Model 15](#_Toc128655348)

[3.2. Execution Schedules And Deliverables 15](#_Toc128655349)

[3.3. Project Communication Model 16](#_Toc128655350)

[3.4. Change Request Management 16](#_Toc128655351)

[3.5. Acceptance Criteria 16](#_Toc128655352)

[4. Commercials 17](#_Toc128655353)

[4.1. Cost for Development 17](#_Toc128655354)

[4.2. Terms and Conditions 17](#_Toc128655355)

# Requirement Details

## Our Understanding of Requirements

There are two components which are integrated with MyForce app, and modules that Sales Reps use during their interaction with HCPs.

**Presentation Manager**

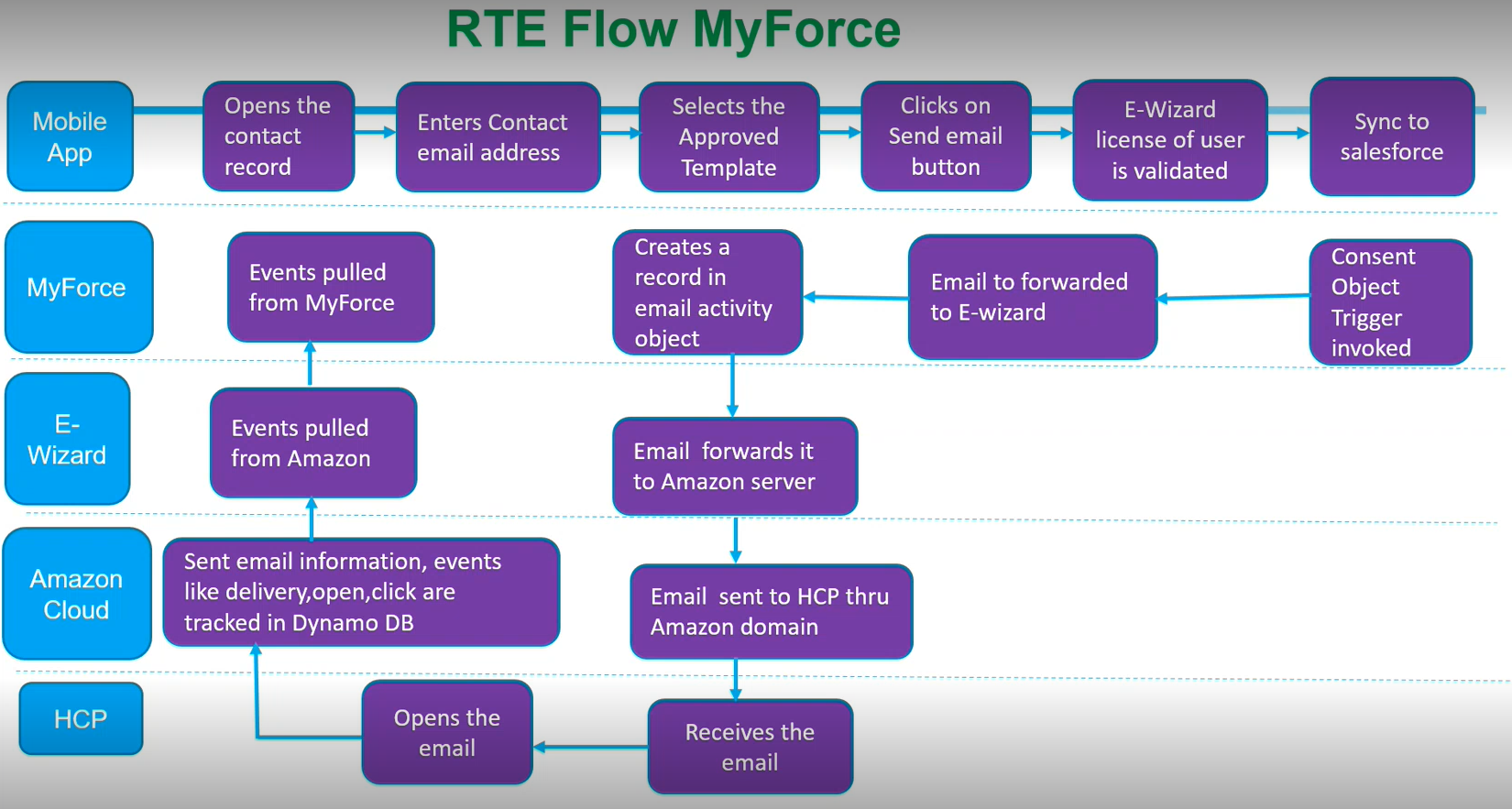
Presentation manager application allows admin users to upload a file in AWS S3 for a presentation record and save the file URL (link) into salesforce Presentation record.

Viatris wishes to replace the existing Presentation Manager application with a home-grown application as the source code of the existing application is owned by 3rd party provider but hosted in Viatris AWS.

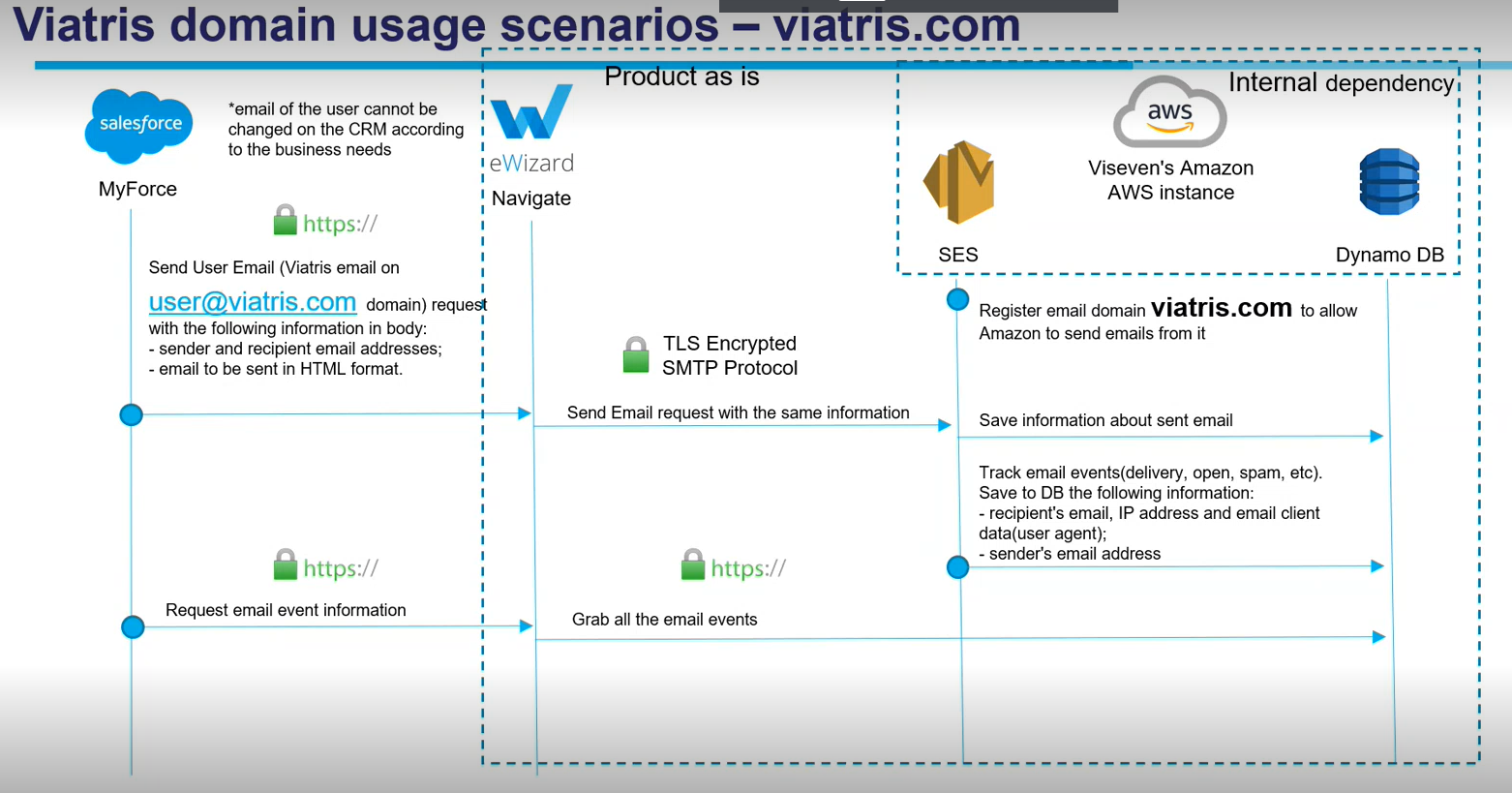
**Viatris Email Agent (Replacement of E-wizard)**

Viatris MyForce RTE (Rep Triggered Email) uses E-wizard, a 3rd party multichannel content experience platform for sending email and pull email event information for Business accounts/Organization Contacts. Reps are expected to have e-wizard license to send any emails. The existing RTE flow in MyForce and E-wizard is depicted below.

Viatris wishes to replace E-wizard tool with a homegrown application. Viatris is looking for an alternative tool /application for E-Wizard which can be deployed on AWS cloud.



*Image 1: This image is a screen grab from the demo recording provided by Viatris. It is used bring clarity in the requirement*



*Image 2: This image is a screen grab from the demo recording provided by Viatris. It is used bring clarity in the requirement.*

### **Functional Requirements**

The following are the features that falls in scope of Presentation Manager and RTE.

**Viatris Email Agent (Replacement of e-wizard navigator module)**

|  |  |  |
| --- | --- | --- |
| Req. ID | App requirements | User Interface/ backend |
| 1 | Users shall be able to login with salesforce username and password. Integration with Salesforce for the login flow. | User Interface |
| 2 | Maintain the mapping of med rep email address to certain other Viatris domain. | User Interface |
| 3 | Replace the med rep email address to other Viatris domain before sending the email | Backend |
| 4 | Send the email content received from MyForce to a 3rd party email service provider | Backend |
| 5 | Pull the email event data from 3rd party email service provider for each email sent | Backend |
| **Changes in existing Integration Points to use Viatris email agent** | | |
| 1 | MyForce would use RTE REST API to send the email content (call triggered from MyForce to RTE) | Backend |
| 2 | MyForce would use RTE REST API to fetch the email event data (call triggered from MyForce to RTE) | Backend |

**Presentation Manager**

|  |  |
| --- | --- |
| **Req. ID** | **Web App Requirement** |
| 1 | User with admin access shall be able to login with salesforce username and password. Integration with Salesforce for the login flow. |
| 2 | Show the presentation records in Presentation Manager user interface using an existing Salesforce Rest API |
| 3 | Allow the admin users to upload the file (Supported file type: zip and max file size :100 GB) for a presentation record. In case a file is already uploaded for a record then an attempt to upload a new file will override the existing file. |
| 4 | Save the uploaded file in AWS S3 bucket. |
| 5 | Generate a signed URL for the saved zip file |
| 6 | Update the salesforce presentation record with the signed URL using an existing Salesforce REST API. |

### Non-Functional Requirements

* Presentation Manager – Max file size – 100 GB
* RTE - Number of emails triggered per day approx. 1100.

## In Scope

* Design and Development of Presentation Manager and Viatris Email Agent (Replacement of e-wizard navigator module)
* Salesforce changes (limited to modifying the existing APIs to integrate with new solution. No new feature development)
* Manual functional testing
* UX design
* CI/CD – Provisioning of 2 environments and basic CI/CD pipeline
* Application Security Testing
* Performance Testing (limited to 2-3 critical services)
* Warranty support (2 weeks)

## Out of Scope

* E-wizard modules
  + Template conversion module (Convert Templates Veeva vault format to MyForce format and push to MyForce (salesforce))
  + Any other modules
* Presentation Manager
  + Changes to existing Salesforce APIs for getting the presentation records and updating the file URL.
* Existing Data Migration (file links for existing presentation files and RTE emails and corresponding events)
* Test Automation (API and UI)
* Security (PEN) Testing

## Supported OS and Browser

Mobile apps will be supported on the following OS versions.

|  |  |
| --- | --- |
| **Device** | **OS Version** |
| Laptop / Desktop | Windows 10/11, Chrome |

## Assumptions

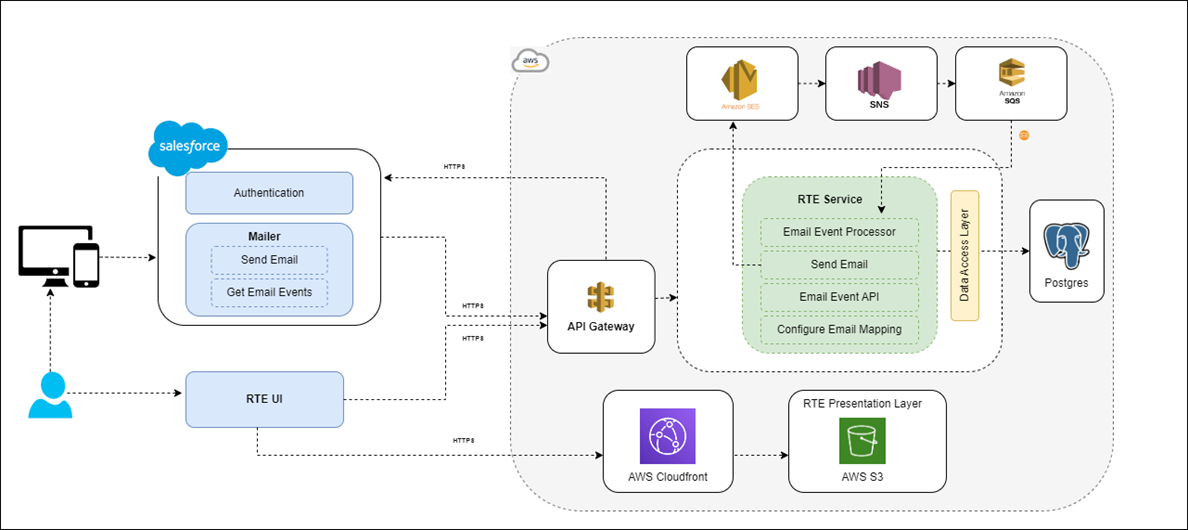
* Salesforce changes for VEA
  + Sending an email & capture the tracking events salesforce rest endpoints and code already implemented, we may need to change it slightly and reuse the same functionality.
  + Email templates is in Salesforce.
  + Med reps will send one email at a time and not the bulk email (Ex. 10-50 email at a time).
  + There will be no changes to email attachment size support available in the existing Salesforce Rest APIs
* Salesforce changes for Presentation Manager
  + Sending the Presentation record and capturing the file URL both the endpoints and code are implemented, we may need to change it slightly and reuse the same functionality.
  + There will be no changes to pagination support available in the existing Salesforce Rest APIs
* VEA UI won’t support email record and associated event tracking/viewing feature.
* Response to Happiest Minds’ queries within two business days. Any delays in response may impact effort and timelines.
* Any other licensed software that arises during the execution of the project will be provided by Viatris or Happiest Minds can purchase and bill it to Viatris after mutual agreement.
* UAT support is limited to a maximum of 2 weeks.
* Estimations are based on our understanding of the requirements as defined in “Requirements Understanding”. Any change in the requirements will have an impact on the timelines and cost.

## Dependencies

|  |  |
| --- | --- |
| **Dependency Factors** | **To be available by** |
| Product owner for the knowledge transition of the application, clarification, and review of the user stories | Start of the project |
| Technical SPOC for feedback and clarifications | Start of the project |
| Provide any coding guideline or standards that happiest minds must follow for development | Start of the project |
| Salesforce sandbox environment with Presentation manager and RTE flow for development and testing. | Start of the project |
| Demo and access of existing E-wizard and presentation manager | Start of the project |
| Code repository access (Git Repo) | Start of the project |
| AWS account | Start of the project |
| Review UAT Test Case document | As needed in the project |
| Feedback on the deliverables for milestone releases | 3 days from the delivery date |
| UAT Testing Completion | Within 2 weeks of the delivery of system tested application |

# Proposed Solution

## Viatris Email Agent Architecture



The above architecture diagram depicts high level architecture of the Viatris RTE solution. The application will be deployed on AWS Cloud.

The following are the major components of the system.

### AWS API Gateway

AWS API gateway is one of the most popular and flexible API gateways from AWS. This is the gateway of the application, all the traffic to access the application passes through it. All the cross-cutting concerns (security, rate limiting, etc.) will be integrated with API gateway. Even though many API gateways are available, we will go with AWS API gateway since our application will be deployed on AWS cloud.

User authentication will be handled in the API Gateway. When the user accesses the presentation layer, the application will make the API call to get the data from backend. All the traffic will be passed through the API gateway. API gateway will authenticate the user before passing the traffic to the backend service. API gateway will connect to salesforce to validate the user. The user will use salesforce credentials to access the application.

### RTE UI

RTE presentation layer will be deployed in AWS S3 bucket along with CloudFront. It is highly available and stores the content at low cost. CloudFront serves content through edge servers and cache the content that will improve the performance of the application.

Web application and backend server will communicate over HTTPS using REST APIs for exchanging data. The proposed solution will use RTE UI to configure the email mapping of the user to send the email from salesforce. RTE backend application will verify and validate the license of the user before sending the email. Any additional features that may be required in future can be incorporated.

The user will use salesforce credentials to access the RTE application.

### RTE Service

The following are the major components of the backend RTE service. Application will be deployed on EC2 instance on AWS. RTE service will have well defined REST APIs which can be consumed by salesforce as well as RTE presentation layer.

### send email

The user can trigger email from the salesforce UI as well as the mobile application. The user can select the template, define the content of the mail, and trigger the email from salesforce or mobile application. Salesforce will make the API call to RTE service to send the email. The mobile application will not directly interact with RTE service to send email. It will proxy through salesforce.

RTE service will take the following actions.

* Validate the user and check the license of the user which is already configured in the RTE service.
* RTE service will interact with AWS SES to send the email.
* Store the reference id and the necessary details in the RTE database.
* We have predefined topic configured in AWS SNS to track the email events. System will capture the following email events only (***sends, failure, rejects, delivered, bounce, spam, delivery delays, subscriptions, open, click***).
* Every event will be pushed to AWS SQS, this will be consumed by the event processor. RTE service will store the event logs as well the latest state (Delivered, Bounced, etc.) as well as user actions (Open, click the link, etc.) of the email.

### Email Event Processor

This module will have the logic of processing the events which are there on AWS SQS. All the events of the email will be pushed to AWS SQS and the event processor will process the events and store it in DB.

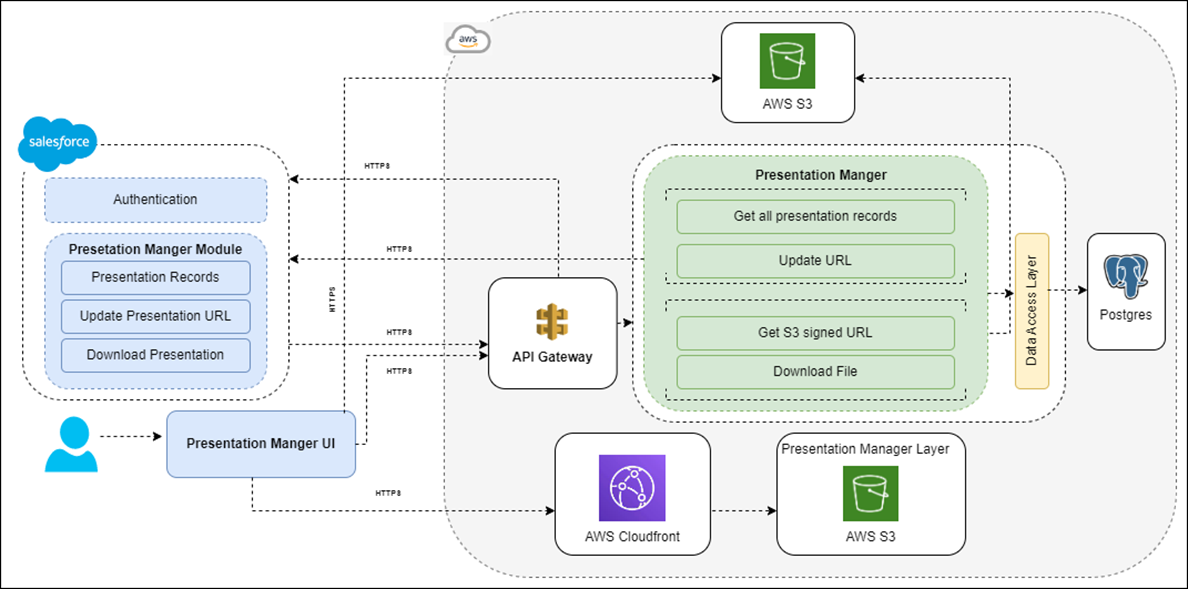
### Email Event API

RTE service will provide the API which will pull the event details for the mail which are stored in the DB. Salesforce will consume the API and store it in salesforce.

### Configure Email Mapping

RTE service will validate the access rights of the user before sending the email. This module will hold the logic of storing the email mapping which will be used to validate the user access.

## Presentation manager architecture



The above architecture diagram depicts high level architecture of the Viatris Presentation Manger solution. The application will be deployed on AWS Cloud.

### Aws API gateway

AWS API gateway is one of the most popular and flexible API gateways from AWS. This is the gateway of the application, all the traffic to access the application passes through it. All the cross-cutting concerns (security, rate limiting, etc.) will be integrated with API gateway. Even though many API gateways are available, we will go with AWS API gateway since our application will be deployed on AWS cloud.

User authentication will be handled in the API Gateway. When the user accesses the presentation layer, the application will make the API call to get the data from backend. All the traffic will be passed through the API gateway. API gateway will authenticate the user before passing the traffic to the backend service. API gateway will connect to salesforce to validate the user. The user will use salesforce credentials to access the application.

### Presentation manager User Interface (UI)

Presentation manager UI will be deployed in AWS S3 bucket along with CloudFront. It is highly available and stores the content at low cost. CloudFront serves content through edge servers and cache the content that will improve the performance of the application.

Web application and backend server will communicate over HTTPS using REST APIs for exchanging data. The proposed solution will use the presentation manager UI to see the list of presentation records which are available in the salesforce. The user will be able to search the presentation record in the UI. If the presentation records are huge in number, we can go with pagination. The user will be able to upload the presentation through the presentation manager. The presentation layer will make the API call to backend service to get the pre-signed URL which will be used to upload the presentation directly to S3 bucket rather than proxying through the server to avoid latency.

File will be split into several parts and the parts will be sent to S3 bucket rather than sending the entire file in one go.

The user will use salesforce credentials to access the presentation manager application. The system will also check whether the user has permission to upload the presentation.

### Presentation manager

The following are the major components of the presentation manager backend service. Application will be deployed on EC2 instance on AWS. Presentation manager service will have well defined REST APIs which can be consumed by salesforce as well as presentation manager UI.

### Get all presentation records

This module will fetch all the presentation records from salesforce. The system will use the existing salesforce API to pull the presentation records based on the filter criteria. This will be rendered in the presentation manager UI. The user can upload the presentation file through the UI based on the permission of the user.

### Get S3 signed URL

The presentation manager UI will directly push the presentation into S3 bucket using the pre-signed URL. This module will generate the pre-signed URL which will be used in the UI to upload the file. System checks the permission of the user before generating the URL.

### Update URL

After successfully uploading the presentation to the repository, the presentation layer will make the API call to the backend to update the uploaded file URL in salesforce. System will make the call to the existing salesforce API to update the URL in salesforce. Store the entry in the DB with the status and AWS S3 URL.

### Download File

When the user clicks to download the presentation from salesforce, it will directly hit the presentation manager API to download the file from S3. The URL which is updated in salesforce will not be the actual S3 URL because of security concerns. All the presentations will not be publicly available on the S3 bucket. Only the application can interact with the repository with the right credentials to download the file. Application will not download the entire file in one go, it will stream the data to download the file.

### Design Considerations

Though these solutions have been presented as two different solutions for RTE and presentation manager, current design considered is single solution. Many components such as Presentation layer, Authentication with salesforce, API gateway, Integration layer for cloud, CloudFront, S3 for deploying the presentation layer, Database, authentication component in salesforce, code setup, logging and monitoring will be reused. Access to these solutions will be based on user permission.

## Testing Approach

### Test Strategy:

Below is the test strategy that would be followed in identifying all the test scenarios to ensure enough test coverage. This also details on the test execution.

### User Scenario Analysis:

User scenarios would be derived based on end user behaviour/interaction with the application. The **Viatris Email Agent & Presentation Manager** is intended to be working on Laptop / Desktop Windows 10/11, Chrome. The features like login with salesforce username & password, on both applications and e-mail validation, uploading files requirements will be tested from end user perspective on the mentioned browser types.

We would be analyzing the business requirements and would design test scenarios to cover all the requirements of login, e-mail triggering, files uploading.

### Test Requirement Analysis

For the Test strategy & planning phases, a systematic approach shall be followed which will help in identifying all the test scenarios, leading to an increase in the effectiveness of test scenarios to detect defects as early as possible in the product development lifecycle.

* Test case design, tools if any will be finalized with Viatris.
* High level milestones on the Testing phases shall be concluded with all the stakeholders.

### Test Cases Development:

* We would be documenting detailed test cases to be used for execution elaborating the Test Scenarios derived from the previous 2 steps. Test cases would be prioritized as P1, P2 based on the criticality of the requirement and segregated as sprint and regression level.
  + P1 test cases include all the positive flow of a particular feature/requirement.
  + P2 test cases include negative flows to validate behavior and error/warning messages thrown by the application.
  + Sprint level test cases include test cases for validation of user stories identified/scoped for a particular sprint.
  + Regression level test cases include end-to-end test cases to cover the user behavior/usage of the application.

### Test Execution & Defect Reporting:

**Test Execution:**

* + - All the test cases would be executed on a Laptop/Desktop with Windows Chrome Browser.
    - 2 rounds of sprint level testing would be conducted to ensure the coverage of the sprint user stories.
    - 2 rounds of regression would also be executed to ensure the end-to-end use cases are covered.

Defects would be categorized as Sev1, Sev2 & Sev3 and would be documented in the agreed upon defect tracking tool. (Sev1 would be like application crash, functionality not working. Sev2 would be functionality has issue but there is a work around. Sev3 would be cosmetic defects like UI issues, error messages are not informative, grammatical errors etc...).

### UAT & Final Sign-Off:

* + Viatris would be preparing and executing UAT testcases all by themselves. Viatris can use the End-to-End Test Cases created by Happiest Minds and tagged as “Happy Paths” to be considered for UAT. Happiest Minds team would provide required support during UAT testing phase. Final Signoff will be obtained from Viatris stakeholders after UAT and fixing & validation of agreed UAT defects.

### Types of testing

* ***Usability testing***:

This type of testing would be considered to cover the usability of Presentation Manager and Viatris Email Agent application on laptop/desktop, we would be validating on the feasibility and ease to navigate and view the application features.

* ***Functional testing:***

The functional testing shall be carried out as per the identified requirements in the requirement documents / brainstorming with the engineering teams and product managers / standards that are planned to be adhered to. In areas where there are enhancements / feature additions / refactoring planned on the core components, QA will study the as-is behavior in any of the existing systems and ensure the as-is functionality is intact wherever the requirements are ported without changes from as-is (Existing) to to-be (new implementation) systems. For areas where there are changes (Modify & Delete) to the existing functionality, QA will study the impact of the change and identify test scenarios accordingly.

* ***Integration and end-to-end testing:***
  + User login - Integration with Salesforce for the login flow.
  + Med rep email address to other Viatris domain
  + Replace the med rep email address to other Viatris domain.
  + Email content received from MyForce to a 3rd party email service provider.
  + Pull the email event data from 3rd party email service provider for each email sent.
  + Changes in existing Integration Points to use Viatris email agent.
  + Integration validation with Admin login
* ***Performance testing:***
  + Following KPIs are considered for Performance Testing:
  + Allow the admin users to upload the file (Supported file type: zip and max file size :100 GB) for a presentation record.
  + Presentation slideshow playing should be < 5 seconds.
  + Apart from this there will be additional tests planned on the client and server side to measure the Throughput and Response times of all the content and pages for varied concurrency.

## DevOps Approach

**In Scope**

* Provisioning two environments in Viatris provided AWS account/Infrastructure.
* Basic pipeline for Dev/Test purpose using AWS Cloud Services
* All PaaS service will be used.

**Out of Scope**

* Parametrized execution
* Automation – Infrastructure (CloudFormation), IaaS, CaaS, Orchestration of Kubernetes
* Backup, Autoscaling, high availability
* Deployment architecture design

## Coding measures

The following code quality measures would be taken care of:

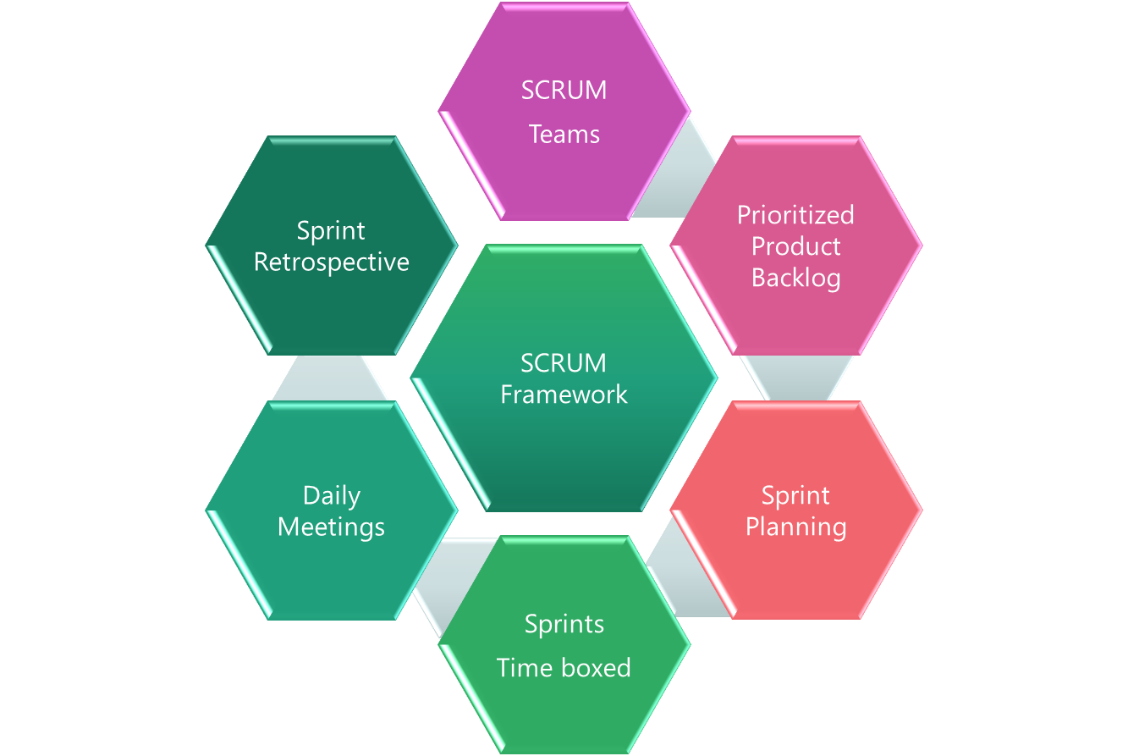
* Unit tests would be written for the features that are in scope of this proposal (would achieve 70% coverage wherever possible)
* Unit test coverage would be integrated with Static code analyzer provided by Viatris.
* Manual code reviews would be done in addition to static code analysis.

## Technology Stack – Indicative

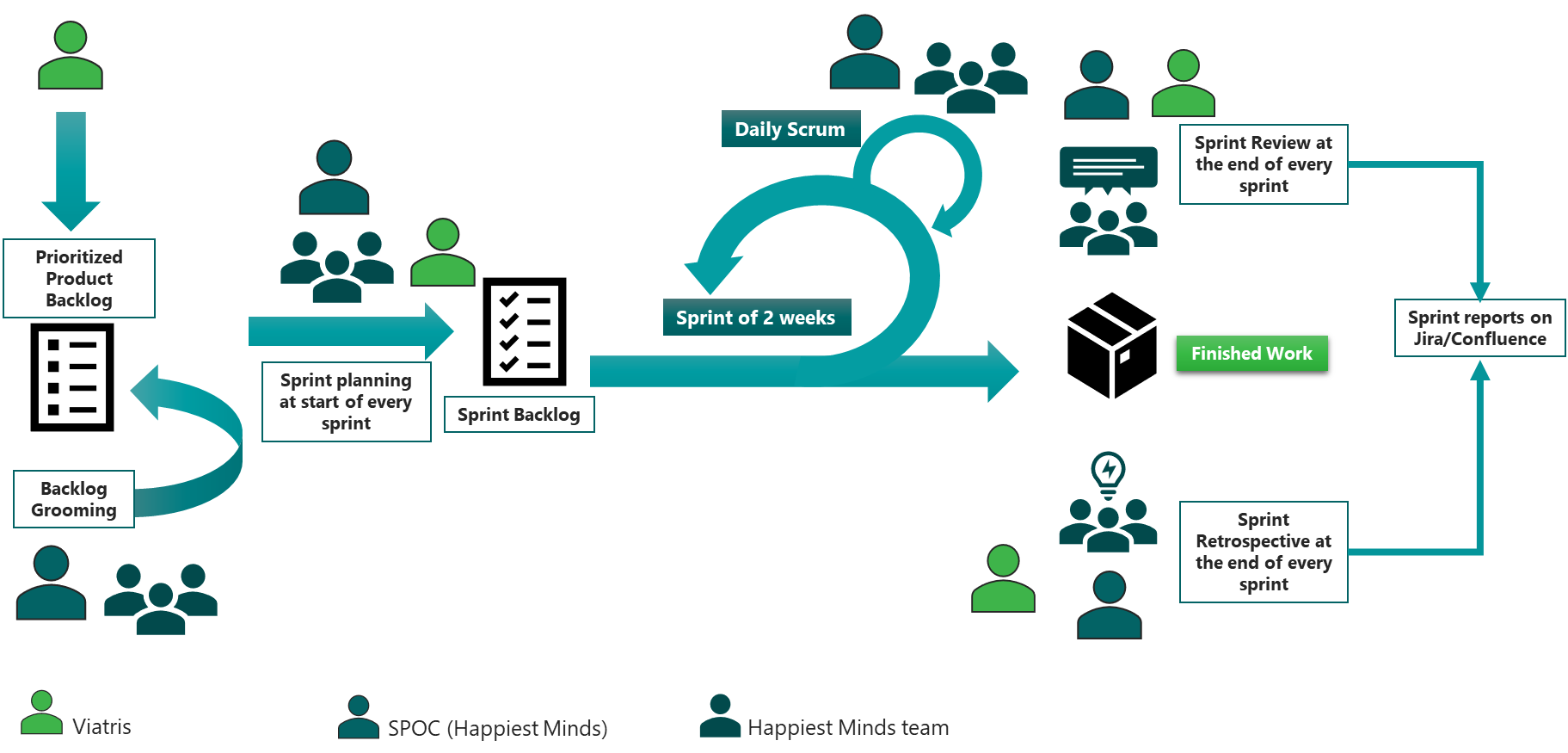
|  |  |
| --- | --- |
| **Feature** | **Technology** |
| User Interface | React, Redux, Jest, S3, CloudFront |
| Cloud | AWS |
| Backend | Java/Node, AWS Lambda, Docker |
| Database | Postgres |
| API Gateway | AWS API Gateway |
| File Storage | AWS S3 bucket |
| Notification & Queue | AWS SES, SQS, SNS |

# Project Delivery Approach

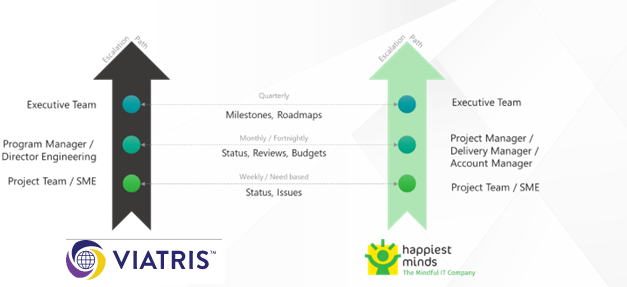
Agile Scrum framework would be used for project execution. Scrum ensures transparency in communication and creates an environment of continuous progress. The below diagram depicts the model followed by Happiest Minds in Agile projects.



2-week sprint cadence would be followed and there would be a demo at the end of each sprint.



## Governance Model



## Execution Schedules And Deliverables

The delivery schedule is based on our current understanding of requirements. Any significant changes in scope of the project are expected to have an impact on the cost and schedule of the project.

The proposed delivery schedule for this project is presented below. The project is estimated to be executed in 8 weeks. It is recommended to execute the project by considering milestone-wise execution. This would help working on the gaps with understanding and identifying the unknown areas. Following are the recommended milestones along with the outcome.

**Assume Project Start Date: (T)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Milestone** | **Due Date in Weeks** | **Owner** | **Deliverables** |
| Project Kickoff | T |  |  |
| Requirement and Solution | T + 1 | Happiest minds & Viatris | * Basic Requirement Document * UX wireframe/ Visuals * Solution Architecture |
| Milestone – 1 | T + 5 | Happiest minds & Viatris | * UX Visuals * Development environment setup * Changes in salesforce * Backend for Presentation & Viatris Email Agent * User Interface for Presentation & Viatris Email Agent * Dev environment Setup and CI/CD * Test Plan and Test Cases * Integration Testing * UAT Test Cases |
| UAT | T + 6 | Happiest minds & Viatris | * Prod environment Setup * UAT Testing and Reporting * Bug fixes |
| Warranty Support\* | T + 8 | Happiest minds & Viatris | * Bug fixes |

*\* Happiest Minds will provide a warranty of 2 weeks and during this period will address any P0 or P1 issues reported on the deliverables. Only defects shall be addressed during this period and any changes/enhancements shall go over the change management process. Post-warranty period, the deliverables are deemed as accepted.*

## Project Communication Model

Happiest Minds proposes the following project-related communication and their frequency:

* Weekly Status Report shall be shared with the Viatris at the end of each week along with the planned set of activities for next week.
* Weekly sync up meeting between the Viatris and Happiest Minds project teams to review the progress of the project

## Change Request Management

Changes to the scope will mean any of the following:

* Any changes to the scope of the project as detailed in section “In Scope”.
* Invalidation of any of the assumptions detailed in section “Assumptions”.
* Any change to the terms and conditions as defined in Commercials and Payment Terms sections.
* Non-fulfillment of any of the dependencies detailed in the sections Dependencies.

In case of change request, the scheduled end date for this Project and/or the fees associated may change. Whenever a change is identified, it will be managed as per the below process:

* For any changes to the scope, either VIATRIS or Happiest Minds will submit a Change Request
* Happiest Minds will issue a Change Order providing the impact of the change to the schedule and/or fees.
* VIATRIS SPOC will review and either approve or cancel the change order.
* Changes will be implemented only after VIATRIS SPOC approval and signing of the change order form by the Parties.
* For any VIATRIS dependencies that are not met or issues that are not resolved, which could impact the schedule – Happiest Minds Project Manager will complete a Change Order and inform the VIATRIS POC.

## Acceptance Criteria

* The UAT test case document would be shared by VIATRIS and signed off 2 weeks before the start of Acceptance Testing
* Acceptance testing shall be performed mutually by Happiest Minds & VIATRIS.
* The acceptance criteria will be passing 100% of the user acceptance test cases with Zero Critical (P0) and High (P1) severity bugs identified.
* Rest of the defects reported during Acceptance testing shall be fixed in Support phase.
* Open bugs would be prioritized and addressed during the Support phase.

**Defect Severity – Definition:** Definition of Defect Severity and Priority are as below.

|  |  |
| --- | --- |
| **Defect Severity** | **Definition** |
| P0 – Critical | Defect may be a showstopper – that is, it stops the user from using the system further. |
| P1 – High | Defect occurs repeatedly and prevents the user from proceeding in the normal way, but a workaround exists. |
| P2 – Medium | A defect is isolated or does not stop the user from proceeding but is annoying and causing inconvenience. |
| P3 – Low | A defect that in no way affects the performance or functionality. E.g.: Aesthetic issues and grammatical errors in messages. |

* The deliverables would be deemed accepted if there is no response/feedback from VIATRIS within 2 weeks post System Tested Application.

# Commercials

## Cost for Development

|  |  |
| --- | --- |
| **Description** | **Amount in INR** |
| Design and Development of Presentation Manager and Viatris Email Agent |  |
| **Total** |  |

**Payment Schedule**

Happiest Minds will invoice VIATRIS starting on Project start date (T) and will follow the payment schedule set below:

*T = Project Start Date*

|  |  |  |
| --- | --- | --- |
| **Milestone** | **Indicate Dates in Weeks** | **Amount (INR)** |
| Project Kickoff | T | 20% |
| Dev-M1 | T + 5 | 60% |
| UAT and Warranty | T + 8 | 20% |

**Payment Terms:** Happiest Minds invoices will be raised at the end of each milestone and will be due for payment 45 days from the date of invoice.

## Terms and Conditions

**Travel:** If any travel and boarding/lodging related expenses are incurred during the execution of the project, it will be charged on actuals to Viatris. Happiest Minds will seek prior permission from Viatris before undertaking any such travel.

**Project specific Costs and Expenses:** Happiest Minds will provide standard PC hardware and software to its team members at its site for execution of work under this SOW. Any project specific specialized hardware, software licenses, testing devices or networking infrastructure required for the project will either be provided by VIATRIS or will be procured and expensed to VIATRIS. Happiest Minds will obtain prior written approval from VIATRIS before procuring or incurring any project specific hardware, software, devices, or network infrastructure expenditure.

**Taxes:** The pricing mentioned exclude GST and any other local and country specific taxes including any withholding tax, as may be applicable