



Release Planning Review (RPR) Briefing Deck

<System/Project Name>
<Release #>

Date:

Presented By:



Release Planning Review Entry Criteria

Artifact	Status	URL
Initial Release Backlog		
Critical Task Schedule		
Capabilities and Constraints		
Team Process Agreement with Definition of Done		
Risk Log		
Security / Privacy Artifacts		
User Acceptance Test Plan		
Section 508 Plan		
Test and Evaluation Plan		



Capabilities, Constraints, and Assumptions

- **Core Capabilities**

<List and describe the core capabilities / objectives for this release>

- **Constraints**

<List and describe the key constraints for this release>

- **Assumptions**

<List and describe any assumptions impacting the release>



Initial Release Backlog & Structure

- **Number of User Stories Identified:**
- **Number of Sprints Planned for this Release:**
- **Number of User Stores planned for each Sprint:**
- **Duration of a Sprint:**
- **Duration of the Release:**
- **Projected Release Readiness Date:**
- **Estimated # of User Stores that can be built in each Sprint:**



Release Process

- **Daily Stand-Up Meetings**
- **Sprint Kickoffs**
- **Sprint Reviews**
- **Sprint Retrospectives**
- **Single Core Product Owner**
- **Team Locations**
- **Daily Stand-Up Meetings**
 - **Daily Stand-Up Meetings at HH:MM AM/PM**
 - **Release Management Daily Stand-Up (Scrum of Scrums) at HH:MM AM/PM**

Core Team Members and POCs

Role	Organization / POC(s)
Product Owner	
IT Project Manager	
Solutions Architect (EA Lead)	
Developers	
Testers	
Business Analysts	
Subject Matter Experts (SMEs)	
IT Security Representative	



Contractors / Key Deliverables

- **Contractor #1**
 - Deliverable Description / Anticipated Delivery Date
 - Deliverable Description / Anticipated Delivery Date
- **Contractor #2**
 - Deliverable Description / Anticipated Delivery Date
 - Deliverable Description / Anticipated Delivery Date
- **Contractor #3**
 - Deliverable Description / Anticipated Delivery Date
 - Deliverable Description / Anticipated Delivery Date



Definition of Done

<Teams should work collaboratively to agree-on and finalize the Definition of Done (DoD) for User Stories: *modify the following as appropriate*>

- The developer verifies that the code is complete and passed unit test.
- The developer has checked the code into source code control.
- Development team has verified that the code passes automated unit tests.
- Independent testers have verified that the story meets the acceptance criteria.
- User Acceptance Testing (UAT) has verified that the story will meet the acceptance criteria in an operational situation. Note that the UAT testers might have additional requested changes that will be entered as feedback and possibly addressed in a later iteration.
- Tests have been checked into configuration management for later use in regression.
- Product Owner sign-off that story meets Definition of Done



Risks and Issues

Date Logged	Type	Description	POC	Status

Core Artifacts

