Following is a suggested BRD template. The template is generic with respect to methodology while providing guidance on tailoring it to specific approaches, such as use-case analysis, structured analysis, UML, and BPMN.

BRD Table of Contents

IT Requirements
 User Requirements
  User Task Overview Diagram
  User Task Descriptions
 IT Service Level (Non-Functional) Requirements
 System State Requirements
  Testing State
  Disabled State
 Static Model
  Static Model: Diagrams
  Multiplicity Rules Table
  Entity Documentation
Test Plan
 Quality Assurance Responsibilities
 QA Standards and Guidelines
 Review and Audit Plan
 Quality Records
 Tools, Techniques, and Methodologies
 Testing Activities
  Preparatory Activities
  White-Box Testing
  "Fit for Purpose" Testing
  Non-Functional Testing
  User-Acceptance Testing
Deployment Plan
 Training
 Conversion
 Scheduling of Jobs
 Rollout
End-User Procedures
Post-Implementation Follow-Up
Other Issues
Sign-Off

Version Control

(Track revisions made to this document in Table 6.1.)

RACI Chart

The RACI chart in this subsection of the BRD describes the roles played by team members and stakeholders in the production of the artifact. RACI is an acronym for Responsible, Accountable, Consulted, and Informed, representing the ways that a stakeholder may be involved with a process or an artifact. The following codes are used in Table 6.2. Note that an additional Supports classification has been added to further clarify roles associated with this document:

*	Authorize	Has ultimate signing authority for any changes to the document.
R	Responsible	Responsible for creating this document.
A	Accountable	Accountable for accuracy of this document (for example, the project manager).
S	Supports	Provides supporting services in the production of this document.
C	Consulted	Provides input.
I	Informed	Must be informed of any changes.

External References

(Table 6.3 lists all other documents referenced in this document.)

Glossary

(List all terms, acronyms, and abbreviations used in this document, and provide definitions or links to entries in the project Glossary.)

Executive Summary

(This is a one-page summary of the document, divided into the following subsections.)

Overview

(This subsection of Executive Summary is a one-paragraph introduction that explains the nature of the project.)

Background

(This subsection of Executive Summary provides details leading up to the project that explain why the project is being considered. Discuss the following where appropriate: marketplace drivers, business drivers, and technology and other drivers.)

Table 6.1 Revision History

Version #	Date	Authorization	Responsibility (Author)	Description

Table 6.2 RACI Chart for This Document

Name	Position	*	R	A	S	C	I

Table 6.3 External References

Document	Location	Publisher	Author

Objectives

(This subsection of Executive Summary details the business objectives addressed by the project.)

Requirements

(This subsection of Executive Summary is a brief summary of the requirements addressed in this document.)

Proposed Strategy

(This subsection of Executive Summary recommends a strategy for proceeding based on alternatives.)

Product/Solution Scope

(Following is a brief description of what is to be included and excluded from the product or solution, divided into the following subsections.)

Included in Scope

(This subsection of Product/Solution Scope is a brief description of the business area and services covered by the product or solution.)

Excluded from Scope

(This subsection of Product/Solution Scope briefly describes business areas and services not covered by the product or solution.)

Constraints

(This subsection of Product/Solution Scope lists predefined requirements and conditions.)

Business Case

(Describe the business rationale for this project and document Critical Success Factors [CSFs]. This section may contain estimates on cost/benefit, Return on Investment [ROI], payback [length of time for the project to pay for itself], market-share benefits, and so on. Quantify each cost or benefit so that business objectives may be measured after implementation.)

Business Services and Processes

(Complete this section if the project involves changes to the workflow of business services and processes. Include all impacted business services and processes, regardless of whether or not they have an IT component.)

Impact of Proposed Changes on Business Services and Processes

(Summarize the impact of changes on the business area in Table 6.4 by identifying the business services and end-to-end processes impacted by the project. If your project employs business use-case modeling, model each end-to-end process as a business use case. If your project is using Structured Analysis, model each as a high-level process.)

Business Service and Process Overview Diagrams

This subsection of Business Services and Processes provides an overview of the business processes impacted by the project. Include diagrams that depict new or changed business services and processes and link them to the actors that initiate them or participate in carrying them out. Include all impacted business services—regardless of whether they have an IT component or not.

Table 6.4 Impact of Proposed Changes on Business Services and Processes

Business Service or Process	[N]ew/ [C]hanged	Desired Functionality	Current Functionality (If a Change)	Stakeholders/ Systems	Priority

If your project is using the business use-case modeling approach, include business use-case diagrams here, indicating business service and end-to-end business processes as business use cases.

If your project is using Structured Analysis, include business-perspective Data Flow Diagrams (DFDs), indicating business services and end-to-end business processes as well as External Entities (actors); model process input and output requirements using data flows.

Business Process Workflow Requirements

This subsection describes the workflow for each business service and process impacted by the project. If necessary, describe existing (As-Is) workflow as well as the new (To-Be) workflow. Workflow diagrams (swimlane workflow diagrams, activity diagrams, BPDs, etc.) are the preferred documentation for cross-functional processes because of the way they visually convey the responsibilities of each participant. (See the BA Toolkit chapter of this book for details on each diagram type.) Diagrams may be augmented with, or substituted by, text, typically using a template included as part of the methodology used on the project.

If your project is using the business use-case modeling approach with UML, document the interaction with the business area textually (for example, using the Business Use-Case Description Template provided in this chapter) and augment with an activity diagram if the flows connect in complex ways. Document the internal workflow used to carry out the process graphically, using activity diagrams with partitions.

If your project employs the business use-case realization modeling element (a UML business modeling extension included, for example, in IBM RUP), distinguish between a business service and the internal processes used to carry it out as follows:

- *Model the business service as a* business use case*. Describe the interaction between the business area and its users (customers) in a business use-case description (referred to in RUP as a business use-case* specification*) using text augmented if necessary with activity diagrams; do not include activities internal to the business.*
- *Model the internal process as a* business use-case realization*. Describe the internal workflow in a business use-case realization description (referred to in RUP as a* specification*) using activity diagrams with partitions.*

If your project is using the BPMN standard, include BPDs. Other diagram types that may by used in this subsection include Block diagrams, flowcharts, and Swimlane Workflow diagrams.

Business Service Level (Non-Functional) Requirements

(This subsection of Business Services and Processes describes Business Service Level Requirements. Service Level Requirements define across-the-board requirements for services and focus on non-functional requirements. For subsections, please refer to the Service Level Requirements Template.

Actors

(This section describes entities that interact with the business area and IT system.)

Workers

(In this subsection of Actors, list and describe stakeholders who act within the business in carrying out business use cases; see Table 6.5.)

Table 6.5 Workers

Department/ Position	General Impact of Project

Business Actors

(In this subsection of Actors, list and describe external parties, such as customers and partners, who interact with the business; see Table 6.6.)

Table 6.6 Business Actors

Business Actor	General Impact of Project

Other Systems

(In this subsection of Actors, list external computer systems potentially impacted by this project. Include any system that will be linked to the proposed system; see Table 6.7.)

Table 6.7 External Systems

System	General Impact of Project

Role Map

(This subsection of Actors provides a centralized easy-to-reference visual summary of actors, depicting the users and external systems that interact with the IT system and their relationships to each other. See "Role Map" in the BA Toolkit chapter of this book for a UML-compliant glossary of symbols and example of a Role Map [limited for the use-case diagram]. If using Structured Analysis, provide a limited version of a DFD depicting only External Entities.)

Business Rules

(List business rules or provide references or links to external business rules documentation. Your business rules may resemble the following example.)

Whenever inventory falls below a trigger level, an automatic order is placed with the supplier.

State Diagrams

(Describe the events that trigger changes of state for key business objects. If your project uses the UML standard, include UML State Machine diagrams here. See "State-Machine Diagram" in the BA Toolkit chapter for more on state diagrams.)

IT Requirements

(This section of the BRD describes requirements originating from the business that the product or solution must fulfill.)

User Requirements

(This subsection of IT Requirements describes requirements for automated processes from a user perspective. If the use-case approach is being employed, model each user task as a system use case. If Structured Analysis is used, model each user task as a Process; place the task in context by defining it as a subprocess of the end-to-end business process it supports.)

User Task Overview Diagram

This subsection of User Requirements provides a graphical overview of user tasks and the user roles that are associated with each task. Each user task represents a piece of meaningful work that a user accomplishes with the assistance of an IT system and is typically completed in a single interaction. If the use-case approach is being employed, model user tasks as system use cases and include system use-case diagrams depicting the actors that interact with each system use case and the dependencies between use cases. See "System Use Cases" in the BA Toolkit chapter of this handbook for a glossary of symbols and example of a system use-case diagram. If Structured Analysis is being used, include Data Flow Diagrams (DFDs) modeling user tasks as IT processes, and indicating the external entities (actors) that interact with them. Indicate the data that is input and output from each process on data flows. See "Data Flow Diagram" in the BA Toolkit chapter of this handbook for a glossary of symbols and examples of DFDs.

User Task Descriptions

In this subsection of User Requirements, describe each user task by documenting the required interaction between the user and the IT system, but without specifying the design.

(For example, do not include screen design or programming specifications.) If the steps of the interaction connect to each other in complex ways, augment the text with work-flow diagrams, such as flowcharts, activity diagrams, and Business Process Diagrams (BPDs), keeping in mind that the focus of these diagrams is specifically the interaction between the user and the system for this user task and not the entire business process[1] (which might include manual steps, interactions with third parties, etc.)

Develop descriptions incrementally. For example, during Initiation, provide only short descriptions. During Discovery, complete the description for each medium- to high-risk user task. Describe low-risk user tasks informally.

If the use-case approach is being employed, include system use-case descriptions (referred to in RUP as use-case specifications) here. For a suggested template, see the System Use-Case Description Template in this chapter.

If Structured Analysis is being used, consider IPO documentation for each process: The acronym refers to the three sections of the documentation: Input, Output, and Process logic. Process logic describing the user-IT interaction may be documented using text such as pseudo-code (a structured writing style using programming-like constructs) or graph-ically, using a workflow diagram.

IT Service Level (Non-Functional) Requirements

(This subsection of IT Requirements describes IT Service Level Requirements. Service Level Requirements define how well the service must perform, rather than the nature of the interaction and workflow. Service Level Requirements are sometimes referred to by other terms, such as "quality attributes" and "non-functional requirements." For a suggested template, see the Service Level [Non-Functional] Requirements Template provided in this chapter.)

System State Requirements

(This subsection of IT Requirements describes how the IT system's behaviour changes when in different states. Describe the features that will be available and those that will be disabled in each state. Please note the distinction between this subsection of IT Requirements and the prior BRD section, State Diagrams. System State Requirements [this subsection] refer to the states of the IT system [Disabled, etc.] and the features that are active or inactive when it is in difference states, whereas the prior section was concerned with the life cycle and states of business objects central to the project, such as a Regis-tration Application, an Order, and so on.)

---

[1]Despite the term "Business" in "Business Process Diagram," BPDs are useful for documenting any workflow—even where the focus is not on the whole business process, but on the interaction with the IT system, as is the case here—because of their wealth of modeling elements (such as inclusive ORs) for handling a variety of situations.

Testing State

(This subsection of System State Requirements describes what the user may and may not do while the system is in the test state.)

Disabled State

(This subsection of System State Requirements describes what is to happen when the system, or part of a system or IT service, is down. Clearly define what the user will and will not be able to do and Vital Business Functions [VBFs] that must remain operational.)

Static Model

(The static model describes the non-timed based business requirements for the IT system. It includes definitions of business entities [business concepts and object types] and related across-the-board business rules that must be supported by the IT system, such as the rule that each policy may be owned by one or more customers.)

Static Model: Diagrams

(In this subsection of Static Model, include static-modeling diagrams depicting business entities and associated business rules and data requirements. If using the UML standard, include class diagrams here; see "Class Diagrams" in the BA Toolkit chapter of this book for details. If using the data-modeling approach, include Entity Relationship Diagrams (ERDs); see "Entity Relationship Diagram" in the BA Toolkit chapter of this book.)

Multiplicity Rules Table

(Include this subsection of Static Model when presenting to business stakeholders, who are likely to be unfamiliar with static modeling diagrams. Include textual descriptions of the business rules expressed in the static model diagram, converting each entity relationship [association] in the model to two business rules [one for each direction].

Table 6.8 is a template for a Multiplicity Rules Table that expresses business rules regarding multiplicity—the number of business objects that may be related to each other—in textual form. Note that the data-modeling term cardinality *and the UML term* multiplicity *are equivalent, that a data- modeling relationship and UML association are equivalent, and that the terms* entity, class, *and* entity class, *as used in a BA context, are roughly equivalent.[2])*

Your documentation may resemble the example shown in Table 6.9.

---

[2]The term *cardinality* is used in data modeling and in the UML. The term *multiplicity* is a UML term. *Entity* is the data modeling term for the UML term *entity class*. Not all classes in the UML are entity classes, but these are the primary classes of interest to the BA. The term cardinality is used in data modeling and in the UML. The term multiplicity is a UML term.

Table 6.8 Multiplicity Rules Table

Each . . . *(Name the entity class at one end of the association.)*	*(Use a verb phrase to name the association between the entity classes at either end.)*	At least . . . *(Document the minimum multiplicity at the other end of the association.)*	*(Document the maximum multiplicity at the other end of the association.)*	*(Name the entity class at the other end of the association. Use plural form if maximum multiplicity is more than 1.)*

Table 6.9 Multiplicity Rules Example

Each . . .	*(Name the association)*	At least . . .	*(Document the maximum multiplicity at the other end of the association.)*	*(Name the entity class at the other end of the association.)*
Policy	Is owned by	1	Or more	Policy Holders
Policy Holder	Owns	1	And only 1	Policy

Entity Documentation

(In this subsection of Static Model, include text documentation to support entities [classes] that appear in the static model diagrams. Not every entity needs to be fully documented by the BA; do a risk analysis to determine where full documentation would most benefit the project.)

Entity Name: *(Name the entity [class]. Use a singular noun phrase, such as Customer.)*

Alias: *(List any other names by which the entity is known within the business domain.)*

Description: *(Provide a short description of the entity.)*

Example: *(Provide an example of an instance [individual object] of this entity.)*

Primary Key: (List the attribute[s] used to uniquely identify an instance of this class.)

Attributes: (Document attributes [data fields]. See Table 6.10.)

Volume: (Document the maximum number of objects of this type that the system must be able to handle.)

Rate of Growth: (Document the rate at which the number of objects of this type is expected to increase.)

(Table 6.10 is a template for documenting attribute properties, such as the method used to derive an attribute's value [if applicable], its data type, and default display format [e.g., MM DD/YY for a date]. In the Dependency column, document any dependencies the attribute may have on other attributes or conditions. For example, dependencies for the attributes of an Invoice might include that a due date must be on or after an invoice date and that the product code on the invoice must match that used in the inventory system.)

Entity (class): (Name of entity or entity class)

Table 6.10 Attributes Table

Attribute	Derived?	Derivation	Type (Numeric, Date, etc.)	Format	Length	Range	Dependency

Test plan

(These requirements are often described in a separate test plan. If they are not addressed elsewhere, describe them here in the BRD. Your test plan should address the steps required to verify whether the solution is "fit for purpose" [does what it is supposed to do] and "fit for use" [works well under specified conditions].)

Quality Assurance Responsibilities

(In this subsection of Test Plan, identify the responsibilities of the quality assurance personnel and specify who those responsibilities will be assigned to.)

QA Standards and Guidelines

(In this subsection of Test Plan, identify the QA standards and guidelines that will be used on the project and indicate how compliance with these standards and guidelines will be determined.)

Review and Audit Plan

(This subsection of Test Plan establishes the reviews and audits that will be performed on the project, their schedule, and their goal.)

Quality Records

(This subsection of Test Plan describes the quality records that will be maintained for the project.)

Tools, Techniques, and Methodologies

(This subsection of Test Plan describes the specific tools, techniques, and methodologies that will be used for quality assurance.)

Testing Activities

(This subsection of Test Plan should address the following testing activities.)

Preparatory Activities

This subsection of Testing Activities describes activities that must be completed prior to executing (running) tests. The BA supports QA in preparing for "fit for purpose" testing (also known as functional or black-box tests) based on the requirements. BA support may include the design of test cases, test scenarios, and test scripts, and the tracing of tests backward to requirements (such as use-case flows) and forward (from requirements to tests) to ensure complete test coverage of the requirements. Decision tables may be used to identify test scenarios when system actions depend on a number of factors. See "Decision Table/Tree" in the BA Toolkit chapter of this book for more on decision tables.

White-Box Testing

(This subsection of Testing Activities describes white-box testing activities that occur once software changes have been made. White-box testing is conducted by the technical team in order to verify whether programs, fields, and calculations function as specified. The BA or technical team specifies the required white-box testing coverage. The lowest coverage is statement coverage [each statement in the code has been executed at least once during testing]. The highest required coverage is, generally, multiple-condition coverage

[every combination of conditions in every decision has been executed at least once]. Technical tests are usually done at the unit level first [testing of the module that was changed]. These are followed by technical integration tests that test the integration of the changed units into progressively larger subsystems and systems.)

"Fit for Purpose" (Black-Box) Testing

(This subsection of Testing Activities describes "fit for purpose" [functional] testing, also known as black-box testing. The BA or dedicated QA staff administers or supervises "fit for purpose" tests, executed to check compliance of the solution with the requirements. Tests should check that all formulae are calculated properly, and that each flow [basic and alternate flows] for each user task [system use case] is tested. Describe principles and techniques to be used, such as structured testing guidelines and boundary-value analysis. Verify requirements coverage by tracing test cases backward to requirements [such as system use cases and flows]. Use decision tables to verify that all required combinations of test scenarios were tested. Use boundary-value analysis to verify the quality of test data.)

Non-Functional Testing

(In this subsection of Testing Activities, describe system tests executed to check compliance of the solution with service level requirements, such as response-time requirements. Tests should verify that the integrity of the system and data remain intact. For more information on test types, see "Checklist: Test Types" in the Tips and Checklists chapter of this handbook.)

For example, your test plan might include the following tests:

- Service specification testing (also known as functional or black-box testing)
- Service level testing
- Service guarantee testing (tests whether guarantees are met regarding availability, capacity, continuity, etc.)
- Usability testing
- Contract and regulation testing
- Service management testing
- Operational testing
- Regression testing
- Load testing
- Stress testing
- Reliability testing
- Security testing

User-Acceptance Testing

(This subsection of Testing Activities describes User-Acceptance Testing procedures.)

Deployment Plan

(The deployment plan is often described in a separate document. If it is not addressed elsewhere, describe it here in the BRD. In this section of the BRD, describe steps required for deploying the product or service into production.)

Training

(In this subsection of Deployment Plan, specify who is responsible for training, who is to be trained, and how training will be done.)

Conversion

(In this subsection of Deployment Plan, specify steps required to convert to the new system. Your plan should describe existing data that must be converted, programs to be promoted to new releases, the granting of user privileges, and so on.)

Scheduling of Jobs

(In this subsection of Deployment Plan, advise IT operations of any changes to the job schedule, often known as the application or system "run book." Indicate jobs to be added to the production run and their frequency, reports to be printed, and jobs and reports that are to be discontinued or changed.)

Rollout

(In this subsection of Deployment plan, describe procedures for rolling out the solution to the production environment and to end users. The plan should include preparations for production-verification testing [also known as production-validation testing], an important and all-too-often neglected part of the life cycle and one that the BA should be instrumental in. The rollout plan should also ensure that all affected users are advised when the project is promoted.)

End-User Procedures

(Document user procedures for the affected user roles. Distribute to end users and provide hands-on user training.)

Post-Implementation Follow-Up

(Follow up within a reasonable time frame after implementation to ensure that the project is running successfully and that its objectives have been met. Determine whether any further enhancements or changes are needed to ensure success of the solution.)

Other Issues

(Use this section for any other issues not covered elsewhere in the BRD.)

Sign-Off

(In this section of the BRD, stakeholders, solution providers, and others confirm their acceptance of the document.)