



Business Analyst BootCamp

BA Intern Bootcamp

- Understand the role of a Business Analyst (BA) in projects
- Learn how to gather, document, and manage requirements
- Get hands-on practice with BA templates and tools
- Collaborate like a BA in Agile ceremonies



- Complete a Capstone Project to showcase skills

This is a Simple concepts document for the above bootcamp

BA Bootcamp - Simple Concepts & Answers

Let's began with Week 1

Week 1 Topics:

1. What is a BA? Why does this role exist?

The BA helps fill this gap by analyzing processes, identifying needs, documenting requirements, and ensuring clear communication between stakeholders and development teams.

The role exists because businesses often struggle to translate goals, customer requirements, or pain points into actionable, testable, and buildable software requirements.

2. Difference between BA, PM, PO, and QA

Role	Core Focus	Key Responsibilities
Business Analyst (BA)	Requirements & process improvement	Gather and analyze requirements, model workflows, create user stories, bridge communication between business and tech teams.
Project Manager (PM)	Delivery & timelines	Plan, schedule, allocate resources, manage risks, and ensure project completion on time and within budget.
Product Owner (PO)	Product vision & backlog	Define product roadmap, prioritize backlog items, and represent customer/business value in Agile ceremonies.
Quality Analyst (QA)	Testing & quality assurance	Test system functionality, identify defects, and ensure the software meets acceptance criteria and user expectations.

3. BA's role in the Software Development Life Cycle (SDLC)

A **Business Analyst (BA)** supports every stage of the SDLC by ensuring the project meets business goals. They gather and analyze requirements, help design and test the solution, and make sure it's ready for users. After deployment, the BA reviews feedback and suggests improvements. Their main goal is to connect business needs with technical solutions effectively.

4. Overview of TechGrit domains (HealthTech, ConTech, FinTech)

TechGrit operates in three key technology domains:

1. **HealthTech:** Building solutions for hospitals, clinics, and insurers like patient management systems, EHR integrations, and telehealth platforms.
2. **ConTech (Construction Tech):** Digital tools for project tracking, materials management, and contractor coordination.
3. **FinTech:** Platforms for payments, lending, compliance, and data-driven financial insights.

5. Assignment: Write a 1-page summary of the BA role in Agile projects.

[edit?usp=sharing](#)

Week 2

1. Types of Requirements (Business, Functional, Non-Functional)

Business Requirements

High-level goals of the organization.

They explain **why** the project is needed.

Example: Increase online sales by 20 percent this year.

Functional Requirements

Describe **what** the system should do.

They define specific actions, features, or behaviors.

Example: The system should allow users to add items to a shopping cart.

Non-Functional Requirements

Describe **how** the system performs.

They include performance, security, reliability, and usability.

Example: The website should load within 3 seconds

2. Elicitation Techniques (Interviews, Workshops, Surveys, Observation)

Ways a Business Analyst gathers information from stakeholders.

- **Interviews:** One-on-one or group conversations to understand needs.
- **Workshops:** Team sessions to discuss ideas and find solutions.
- **Surveys or Questionnaires:** Collect feedback from many users.
- **Observation:** Watch users perform tasks to learn how processes work in real life.

3. Prioritization Methods (MoSCoW, RICE)

MoSCoW Method

This method helps decide which features or tasks are most important in a project.

- **Must Have:** These are very important and must be included for the project to succeed.
- **Should Have:** These are important but not critical. The project can still work without them for now.
- **Could Have:** These are nice to have if there is enough time and resources.
- **Won't Have:** These will not be done in the current phase but can be added later.

RICE Method

This method helps rank features using four factors:

- **Reach:** How many people will use or benefit from it.
- **Impact:** How much value it brings to users or the business.
- **Confidence:** How sure we are about the impact and effort estimates.
- **Effort:** How much time and resources it will take.

The RICE score is calculated as **Reach × Impact × Confidence ÷ Effort**.

A higher score means a higher priority.

4. Writing Good User Stories (INVEST Model)

INVEST Model for User Stories

The INVEST model helps Business Analysts make sure each user story is clear and ready for development. Each letter in INVEST represents a quality that makes a user story strong and effective.

I → Independent

Each story should stand on its own and not depend on another story to work. This makes it easier for the team to develop and test it separately.

Example: As a customer, I want to reset my password so that I can access my account if I forget it.

Not Independent: As a customer, I want to reset my password only after the profile update feature is done.

N → Negotiable

A user story should be open for discussion and improvement. It is not a fixed rule but a starting point for conversation between the team and stakeholders.

Example: As a student, I want to view available courses so that I can plan my schedule.

Not Negotiable: As a student, I want a course list that must look exactly one design.

V → Valuable

Each story should provide real value to the user or the business. It should clearly show why it is useful.

Example: As a user, I want to receive email notifications so that I know when my order is shipped.

Not Valuable: As a system, I want to record logs for background tasks.

E → Estimable

The story should be clear enough for the team to estimate how much effort or time it will take.

Example: As an admin, I want to delete inactive accounts so that the system stays clean.

Not Estimable: As a user, I want the app to be perfect and user friendly.

S → Small

The story should be small enough to finish within one sprint. If it is too big, it should be divided into smaller parts.

Example: As a customer, I want to add items to my cart so that I can buy multiple products together.

Too Large: As a customer, I want a complete shopping experience with cart, wishlist, reviews, and delivery tracking.

T → Testable

A good user story must have clear acceptance criteria so that it can be tested and verified.

Example: As a teacher, I want to download student reports so that I can review performance offline.

Acceptance Criteria:

1. Teacher can download reports in PDF format.
2. File name includes the student name and date.
3. The downloaded report matches the online version.

5. Assignment: Write 5 user stories with acceptance criteria for a sample project.

How to Think About It in Simple Terms

- User Story = **What the user wants** (the "wish").
- Acceptance Criteria = **How we know it works** (the "proof").

Example:

- *User Story:* "I want a pizza delivered."
- *Acceptance Criteria:*
 - Pizza arrives hot.
 - Pizza matches the order (size, toppings).
 - Delivered within 30 minutes.

If all 3 are true → the story is complete .

These are the User Stories for the project

Week 2 Assignment

1. Donor – Library Needs

As a donor, I want to check which library needs most of the books so that I can prefer that one first.

Acceptance Criteria:

- Donor can view a list of all libraries.
 - Each library shows number of pending/needed books.
 - Donor can sort or filter libraries by "most needed."
 - The list updates regularly (e.g., daily or in real-time).
-

2. Donor – Donation Tracking

As a donor, I want to see the activity of my purchase so that I will know that this is legit.

Acceptance Criteria:

- Donor receives confirmation when the purchase is made.
 - Donor can track donation status (pending, shipped, delivered).
 - Library/admin uploads proof of delivery (e.g., photo).
 - Donor can view donation history in their account.
-

3. Librarian – Wishlist Management

As a librarian, I want to add the books which are needed so that donors can donate the specific books.

Acceptance Criteria:

- Librarian can log in to their account.
 - Librarian can add/edit/remove books in the wishlist.
 - Wishlist items include title, author, quantity, and link (Amazon/Flipkart).
 - Donors see updated wishlist immediately after changes.
-

4. Donor – Cart Functionality

As a donor, I want to add books in a cart so that I can do multiple purchases.

Acceptance Criteria:

- Donor can add multiple books to the cart.
 - Cart displays total items, quantity, and price.
 - Donor can remove or update items before checkout.
 - Cart is saved until purchase is complete.
-

5. Admin – Library Registry

As an admin, I want to add a library to the registry so that donors can see and donate.

Acceptance Criteria:

- Admin can create new library profiles with details (name, location, contact).
- Admin must verify a library before it becomes visible to donors.
- Donors only see verified libraries in the registry.
- Admin can update or deactivate libraries if needed.

WEEK 3

This is based on the Library Donor project

[Project.pdf](#)



Jira / Azure DevOps → creating & managing user stories

This is the link for Jira

<https://hemanthofficial666.atlassian.net/jira/software/projects/LC/boards/67/backlog?atlOrigin=eyJpIjoiZTE3MTM4MmZjMWE1NGYxMmFkYjUwMjBIMWUxN2M3MjYiLCJwIjoiaiJ9&epics=visible&issueParent=1000000000000000000>

I USED MIRO FOR MY PROJECT

wireframes

https://miro.com/welcomeonboard/OCtSa3F0cWowV0YrbII4eVZyV0xxcmM5czRSdGFnNVdpdmxYYIIValZBZnNDdnZkYnBkc/share_link_id=786345307195

Figma / Balsamiq → wireframes basics

due to some reasons I am having an error (when you click on something its not taking you to the page)

I am willing to screen sharing my figma project

<https://www.figma.com/make/FsTopIkaD8GeYzIZ7ynn8M/Donor-Screens-Design?node-id=0-4&t=HRj3DqgxtaVROkmM-1>

Requirements Traceability Matrix (RTM)

Requirement ID

Each requirement has a unique number to identify it easily.

Requirement Description

A short and clear explanation of what the requirement is.

Source

Shows who or where the requirement came from, such as a client or stakeholder.

Functional Requirement or Design Reference

Shows how the business need connects to the system or design solution.

Test Case ID

Links the requirement to the test case that checks if it works correctly.

Status

Shows the progress of the requirement, for example Not Started, In Progress, or Completed.

Priority

Tells how important the requirement is, such as High, Medium, or Low.

Owner

The person responsible for completing or reviewing the requirement.

Comments

Used to write any notes, updates, or feedback related to the requirement.

WEEK 3 ASSIGNMENT

[edit?usp=sharing](#)

WEEK 4 Assignment

1. Agile Ceremonies

Agile projects have five main ceremonies that help teams plan, work, and review progress.

Daily Standup:

A short meeting where each team member shares what they did yesterday, what they will do today, and if there are any issues.

Backlog Grooming (Refinement):

A session where the team reviews the product backlog. The Business Analyst helps to make sure all user stories are clear, updated, and ready for the next sprint.

Sprint Planning:

The team decides what tasks will be completed in the next sprint. The Business Analyst helps explain requirements and answer questions from developers and testers.

Sprint Demo (Review):

At the end of the sprint, the team shows the completed work to the client or stakeholders. The Business Analyst helps collect feedback and confirms if requirements are met.

Retrospective:

A meeting where the team discusses what went well, what can be improved, and how to work better in the next sprint. The Business Analyst listens and suggests process improvements.

2. Business Analyst Role in Each Ceremony

- **Standup:** Shares updates on requirements and supports the team with clarifications.
- **Backlog Grooming:** Reviews user stories, updates acceptance criteria, and ensures items are ready for development.
- **Sprint Planning:** Explains priorities and helps define clear sprint goals.

- **Demo:** Confirms that features meet business needs and collects feedback.
 - **Retrospective:** Helps identify ways to improve communication and requirement clarity.
-

3. Working with Developers, QA, UX, and Clients

The Business Analyst works as a bridge between all groups.

- With **Developers:** Provides requirement details and clears doubts.
 - With **QA Testers:** Helps create test cases and verifies that acceptance criteria are met.
 - With **UX Designers:** Reviews layouts and ensures the design matches user needs.
 - With **Clients:** Gathers feedback, clarifies changes, and confirms that business goals are met.
-

4. Handling Scope Changes and Requirement Updates

In Agile, changes can happen anytime. The Business Analyst:

- Reviews the new request and checks its impact on time and cost.
- Updates the user stories and backlog.
- Communicates the change clearly to all team members.
- Works with the Product Owner to decide if the change fits in the current sprint or next one.

WEEK 4 ASSIGNMENT

I got this from [claude.ai](#) it give us an understanding how the BA works in agile

<https://claude.ai/public/artifacts/05aebd8c-f540-4de8-afde-8056edf901ce>

WEEK 5 Capstone Project

[Project.pdf](#)

 [Project Library](#)

 [PROJECT NOTES](#)

1. Business Goals

- Build a **community impact platform** connecting donors (urban, NRI, local) with specific causes.
 - Launch the **first modular MVP: "Library Connect"** — a platform enabling donors to send books or funds to village/community libraries.
 - Ensure **transparency and measurable impact** through verified updates and delivery confirmations.
 - Lay the groundwork for an **umbrella ecosystem** that can later expand into other social causes (education, hygiene, scholarships, etc.).
-

2. Key Stakeholders

Stakeholder	Role	Responsibilities
Donors	Primary Users	Browse libraries, donate books/funds, receive delivery/impact updates.
Libraries / Coordinators	Beneficiaries	Post needs, confirm delivery, share proof and stories.
Admins	Platform Operators	Verify libraries, validate donations, manage disputes, oversee impact reporting.

3. Pain Points

- Lack of **centralized, verified platforms** for small community donations.
- Donors lack **trust** and transparency in how contributions are used.
- Libraries face **limited visibility** and inconsistent support.
- Manual donation processes cause **communication delays** and **no delivery confirmation**.

4. MVP Features (MoSCoW Prioritization)

Must Have

- Library registry with verified profiles.
- Wishlist posting for required books.
- Donor fulfillment (via Amazon/Flipkart or fund pledges).
- Delivery confirmation with photos/status updates.
- Admin verification and approval workflow.

Should Have

- Donor dashboard (history, impact stats).
- Email/SMS notifications for donation and delivery events.
- Simple analytics (books donated, libraries supported).

Could Have

- Recurring donations (monthly).
- Social sharing of completed donations.
- Multi-language support (English + Telugu/Hindi).

Won't Have (for MVP)

- Multi-cause integration beyond libraries.
- Gamification or reward system.

5. High-Level Functional Overview

1. **Library Registry** → searchable directory with profiles, photos, needs.
2. **Wishlist Management** → structured, library-posted book requests.
3. **Donation Flow** → donors fulfill needs through direct purchase or fund pledges.
4. **Delivery Tracking** → coordinators confirm receipt with photo proof.
5. **Impact Updates** → donors receive real-time notifications upon completion.

6. Success Metrics (KPIs)

- Number of libraries registered and verified.
 - Number of completed donations.
 - Average books donated per month.
 - Ratio of fulfilled vs pending requests.
 - Donor satisfaction rate and engagement feedback.
-

7. BA Deliverables / Visuals

- **System Diagram:** Donor → Platform → Library → Delivery → Impact.
- **Wireframes:** Donor, Librarian, Admin views.
- **User Journey Map:** e.g., NRI Donor → Select Library → Donate → Delivery Confirmed → Receives Impact Story.

BRD :

[Library_Connect_BRD.pdf](#)

PRD :

[Library_Connect_PRD.pdf](#)

USER STORIES : THIS IS FROM WEEK 2

1. Donor – Library Needs

As a donor, I want to check which library needs most of the books so that I can prefer that one first.

Acceptance Criteria:

- Donor can view a list of all libraries.
 - Each library shows number of pending/needed books.
 - Donor can sort or filter libraries by "most needed."
 - The list updates regularly (e.g., daily or in real-time).
-

2. Donor – Donation Tracking

As a donor, I want to see the activity of my purchase so that I will know that this is legit.

Acceptance Criteria:

- Donor receives confirmation when the purchase is made.
- Donor can track donation status (pending, shipped, delivered).
- Library/admin uploads proof of delivery (e.g., photo).

- Donor can view donation history in their account.
-

3. Librarian – Wishlist Management

As a librarian, I want to add the books which are needed so that donors can donate the specific books.

Acceptance Criteria:

- Librarian can log in to their account.
 - Librarian can add/edit/remove books in the wishlist.
 - Wishlist items include title, author, quantity, and link (Amazon/Flipkart).
 - Donors see updated wishlist immediately after changes.
-

4. Donor – Cart Functionality

As a donor, I want to add books in a cart so that I can do multiple purchases.

Acceptance Criteria:

- Donor can add multiple books to the cart.
 - Cart displays total items, quantity, and price.
 - Donor can remove or update items before checkout.
 - Cart is saved until purchase is complete.
-

5. Admin – Library Registry

As an admin, I want to add a library to the registry so that donors can see and donate.

Acceptance Criteria:

- Admin can create new library profiles with details (name, location, contact).
- Admin must verify a library before it becomes visible to donors.
- Donors only see verified libraries in the registry.
- Admin can update or deactivate libraries if needed.

WIREFRAMES

I USED MIRO FOR MY PROJECT

https://miro.com/welcomeonboard/OCtSa3F0cWowV0YrbII4eVZyV0xxcmM5czRSdGFnNVdpdmxYYIIValZBZnNDdnZkYnBkc/share_link_id=786345307195

Figma / Balsamiq → wireframes basics

due to some reasons I am having an error (when you click on something its not taking you to the page)

I am willing to screen sharing my figma project

<https://www.figma.com/make/FsTopIkaD8GeYzIZ7ynn8M/Donor-Screens-Design?node-id=0-4&t=HRj3DqgxtaVRQkmM-1>

RTM

<https://docs.google.com/spreadsheets/d/1wUM73eEqhXJRJIYfIYZXPK1EaFgiB2J76worc3P1PaY/edit?gid=0#gid=0>

THANK YOU