1. Lean Startup

- methodology for developing businesses & products
- shorter development cycles
- discover if proposed business model is viable

1.1 Business-Hypothesis-Driven Experimentation

1.2 Iterative Product Releases

1.3 Validated Learning

2. Minimum Viable Product (MVP)

- version of product with just enough features to be usable by early customers
- to get early feedbacks for product viability & future improvements
- early release = avoid lengthy, unnecessary work
- Client interaction with product is more important than asking for their thoughts on it
- gain understanding on customer interest on product
- spend less effort & cost if product not viable

3. User Stories

- requirements are gathered as user stories in Agile model
- Dev team discuss with product owner / clients on requirements
- Dev team divide work and converts requirements into user stories
- Each user story should contribute to value of overall product
- Each story will have **User Acceptance Criteria** and should meet it 100% before story can be closed

> ## Creating User stories (Templates):
>
> ###1. Role-Action-Benefit (most common) As a <Role>, I want to perform <Action> so that I can <Benefit> eg: As a <Bank Customer>, I want to <withdraw money from ATM> so that <I am not constrained by opening hours or queues at banks>
>
> ## INVEST guidelines

> **Independent**
>
> - each user story should be independent
>
> **Negotiable**
>
> - user stories are not detailed contracts
> - scope can be negotiated
>
> **Valuable**
>
> - every story should add value to product
>
> **Estimable**
>
> - story written is a way understandable to dev
> - dev have idea on how to implement & how long it will take
> - properly sized story
>
> **Testable**
>
> - stories should be testable to help determine if a story is completed or not
> - should have valid acceptance criteria (objective, specific, testable, measurable)
>
> ### CCC (Card, Conversation, Confirmation)

4. Burndown chart

- graph of work left to do vs time
- y-axis = work left to do
- x-axis = time
- typically used with Scrum but can actually be used with any project to track progress over time
- work left can be in terms of:
  - story points
  - number of stories
- Number of workdays, workers & efficiency factor of worker also required
- Time required for project = [(Number of workdays) / (num of workers)] / worker efficiency