|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Online | | | | |
|  |  | | |  |
| buy ticket | | | | |
|  | | CINEMA APP |  | |

TABLE OF CONTENTS

[agenda 3](#_Toc68847307)

[1. Brief 3](#_Toc68847308)

[2. Description Scope 4](#_Toc68847309)

[3. Financial Scope 5](#_Toc68847310)

[4. Focused on Delivery 5](#_Toc68847311)

[5. Project Execution 7](#_Toc68847312)

[1. DESIGN 8](#_Toc68847313)

[1. PERSONA 8](#_Toc68847314)

[2. Requirement analysis(non/functional) 9](#_Toc68847315)

[3. UML Representation analysis 12](#_Toc68847316)

[4. Use Case 13](#_Toc68847317)

[5. Class Diagrams 14](#_Toc68847318)

[6. State machine/State chart Diagram 15](#_Toc68847319)

[7. Storyboards & Wireframes (low fidelity) 16](#_Toc68847320)

[8. User Flows 17](#_Toc68847321)

[9. Web Site map Design & Planning 18](#_Toc68847322)

[10. Heat Map/mockup? 20](#_Toc68847323)

[2. Project management 21](#_Toc68847324)

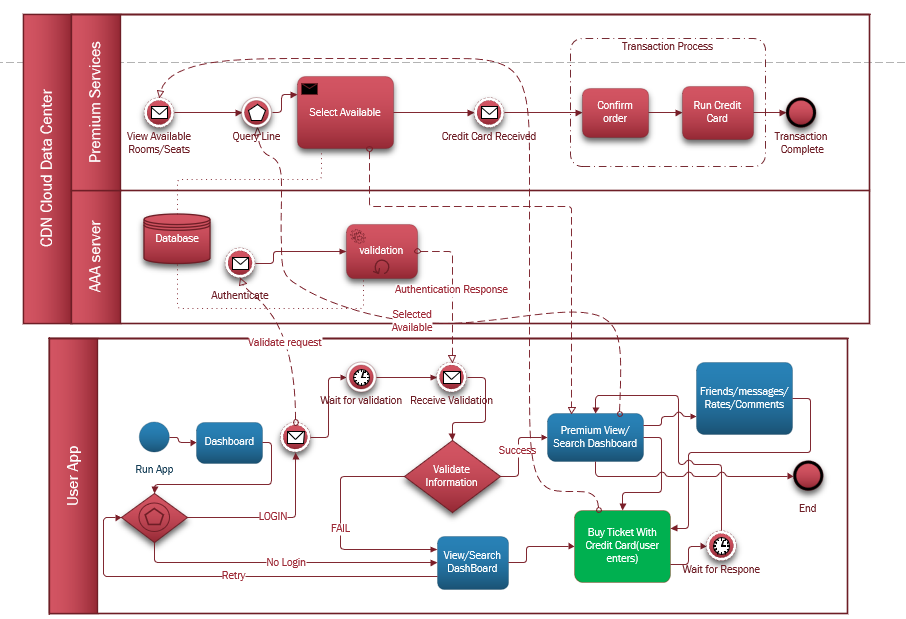
# agenda

## Brief

**CINESOCIAL – Product Design (UI/UX back-end) Project**

* Project Name:
  + CINESOCIAL
* Project Description:
  + A client whishes to develop and test a platform to enable online ticket booking system for cinemas, to host their own content on their own domain.
* Who is this for?
  + Everyone – this product is for everyone who wishes to learn about cinema or just entertain him/herself by buying premiere tickets to watch fast on demand movies.
* Feature List (product requirements)
  + Easy onboarding
  + Example | Pricing | login | Sign up
  + No monthly fees
  + Personalization screen
  + Ticket booking
  + Friend System like social media
* Competitors & Product Inspiration
  + IMDb
  + Gamato
* Deliverables
  + Wireframes for client approval
  + High fidelity prototype of the product
  + User Testing
  + Usability Report
  + UI Assets for developers
* Cost
  + $4000 total with $93 p/hour for any additional work outside the brief
  + 50% payment required to begin work
* Timeline/Deadline
  + The duration is described below as subtasks

## Description Scope



BPMN 2.0 Visual Process Representation of Description Below(**activity-like**)

The purpose of this document is to provide valid information for its members including Low level Design and High-Level design in a concise application of the Business Cinema Content application.

The application will consist of front & back-end interfaces functionality. It will allow the guest user to see list cast actors, premiere trailers, opening hours, days and rooms by making online reservations on-click but will have VIP registered users who can make rating, comments, to have followers, e.g., similar features of a social media limited mode & cinema interactive. The application will be a subset of an information system that will interact with users and other database systems.

## Financial Scope

The project will observe deadlines, Budget, vision of stakeholders.

* Investment decision
* Working Capital decision
* Financing decision
* Dividend decision
* Ensures liquidity
* Profit management

## Focused on Delivery

**Subtasks/activities.**

1) Requirements analysis

2) Database design

3) Interface design

4) Database development and data entry

5) Interface development

6) System completion

7) Application installation and control (integration)

8) Training and acceptance/support

**The duration of the subprojects is analyzed below:**

subtask 1: 1 month/s (sprints)

subtask 2: 1 month/s (sprints)

subtask 3: 0,5 month/s (sprints)

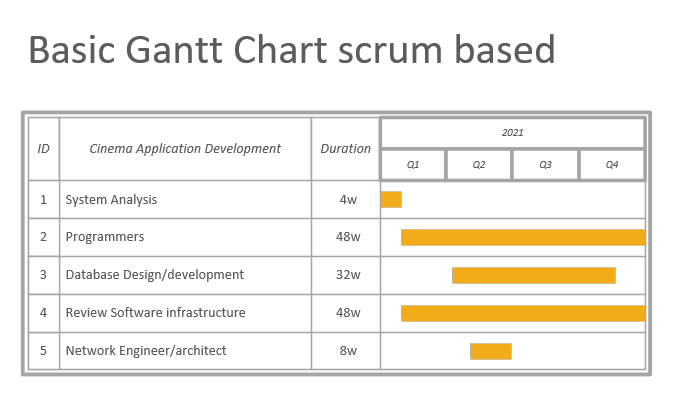
subtask 4: 3 month/s (sprints)

subtask 5: 2 month/s (sprints)

subtask 6: 1 month/s (sprints)

subtask 7: 1 month/s (sprints)

subtask 8: 0,5 month/s (sprints)



Estimated final plan

## Project Execution

Approaches available: procedural/waterfall, agile, Dev ops

A base of staff specialization and available infrastructure is selected agile scrum team approach. The main course recording and management will become a means of collaboration & agile platform JIRA with tracking and problem recording system/tickets source & version control DVCS GIT.

**Product BACKLOG**

Below is an initial estimate which will change dynamically (updated / passage of time) with more information as the priorities will change as well as the requirements of the project may change.



\*from the above section Agenda everything is abstracted including the Gantt and Product backlog these are not in any means any close to actual requirements but a close generalized estimate of what to be expected. Latter will be introduced actual implementations in this document.

# DESIGN

## **PERSONA**



The project will expand as what NIKI wants.

## **Requirement analysis(non/functional)**

**indicative specification functional requirements**

**Function Name:** Account login

**Description**: User admin login portals

**INPUTS**: Front-end portal authenticated in back-end.

**System behavior**: The system takes Data comparing each unique user individually based on their input fields.

**OUTPUT**: logged in profiles on site.

**Function Name:** Feedback

**Description**: Experiences and complaints comments.

**INPUTS**: Hand written from the user.

**System behavior**: The system takes complaints, notes or anything related from a textfield and sends it in the Database for further logging.

**OUTPUT**: Admin access logging.

**Function Name:** Scheduled events

**Description**: Available on Season to watch movies.

**INPUTS**: No user inputs just most recommended and demanded movies on cycle.

**PROCESS**: The system represents top watched movies along time and categorizes them on cycle based.

**OUTPUT**: Front-end on-site results.

**Function Name:** Ticket Booking

**Description**: User buys tickets admin handles them based on available Seats.

**INPUTS**: From Front-end source data buying tickets bound to or not to an account.

**PROCESS**: The app checks if a user is logged in or not and keeps buy ticket record accordingly by sending them in the Database storage. The admin is able to setup Theatre, add movies and arrange seats. Accept or deny tickets already bought from customer. The Data stored in Database with requests/fetch Data is being displayed accordingly to panels layout.

**OUTPUT**: Reserved tickets are being fetched from database and been displayed in admin portal for further support/management.

**Function Name:** Movie setup

**Description**: Add on the fly new on demands movies.

**INPUTS**: Front-end page admin account.

**PROCESS**: The system is capable of capturing demanding movies from admin page and inserts them into the database. Real time update with front end users to see.

**OUTPUT**: Front-end User pages.

**Function Name:** Movies Info

**Description**: Information and news about a movie.

**INPUTS**: from admin database

**PROCESS**: This system behaviour accepts inputs from admin and shows information accordingly about information, casting, trailers.

**OUTPUT**: Front-end result page

**indicative specification non-functional requirements**

**Function Name:** Portal differences on login

**Function Name:** Fast and responsive Database logging system

**Function Name:** Realtime updates

**Function Name:** ssl(https), pki connections

**Function Name:** Portal differences on login

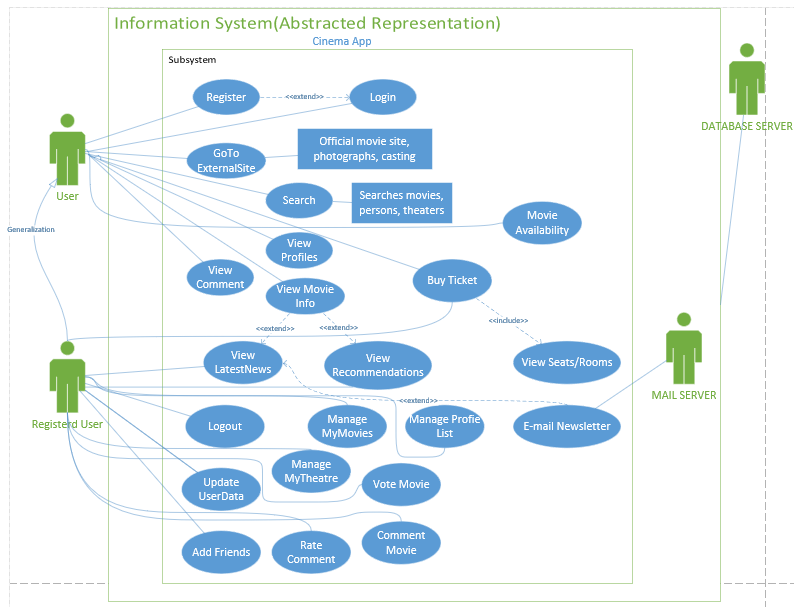
**Function Name:** Load balancing network server offload

**Function Name:** Data center private cloud setup

**Function Name:** Esxi Vmotion enabled

**Function Name:** CIA (Confidentiality, integrity and availability) triad

## **UML Representation analysis**



## **Use Case**

**Use Case**: Login portal

**Main user**: admin, user

**Short Description**: User or admin enter credentials and gets checked against database queries. The inputs fields apply security rules.

**Conditions:** only registered users can use the system with Premium Services.

**Exit status**: Successful Cinema Booking system for users.

**Main Flow**:

1. User registers
2. System checks input integrity [Alternative Flow A]
3. User Logins
4. System Checks input integrity [Alternative Flow B]
5. Enjoy movies

**Alternative flows**

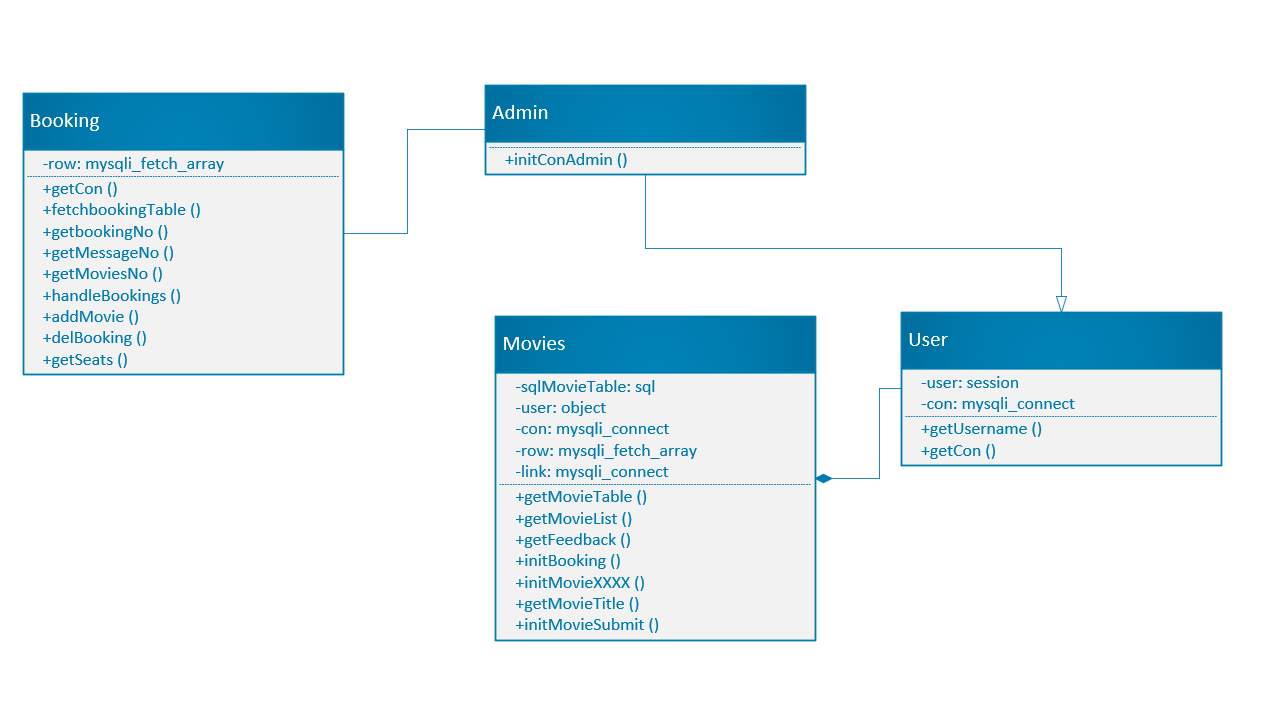
Alternative flow A: Field checking

If not valid input check jumps on Step 1

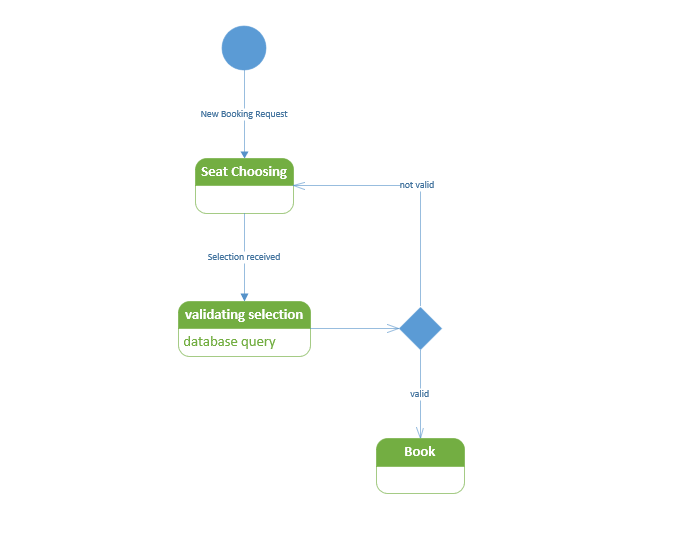
Alternative flow B: Field checking

If not valid input check jumps on Step 3

## **­Class Diagrams**

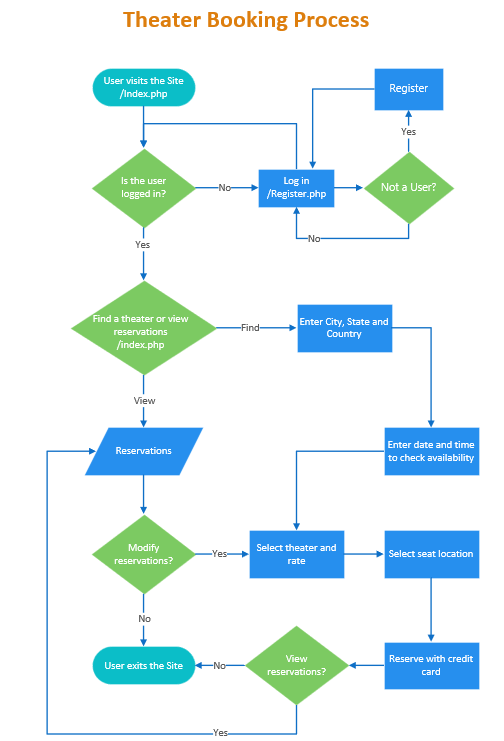


## **State machine/State chart Diagram**

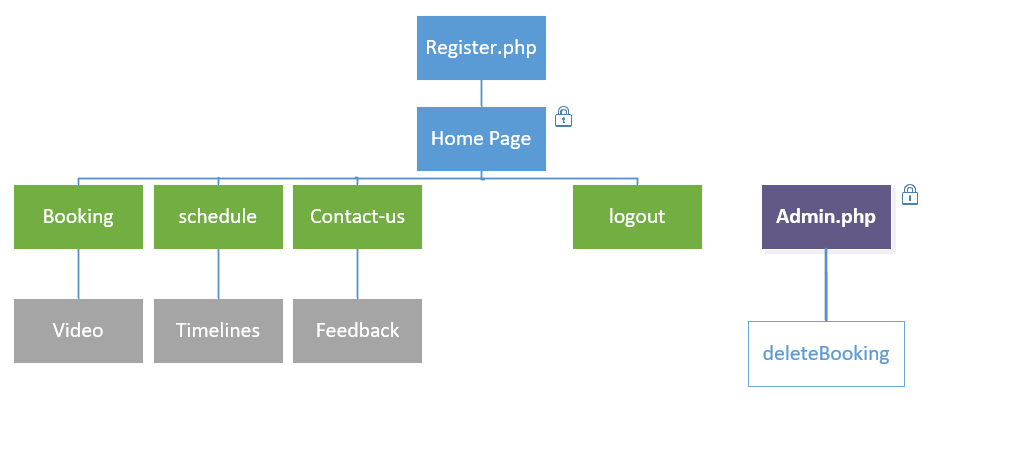


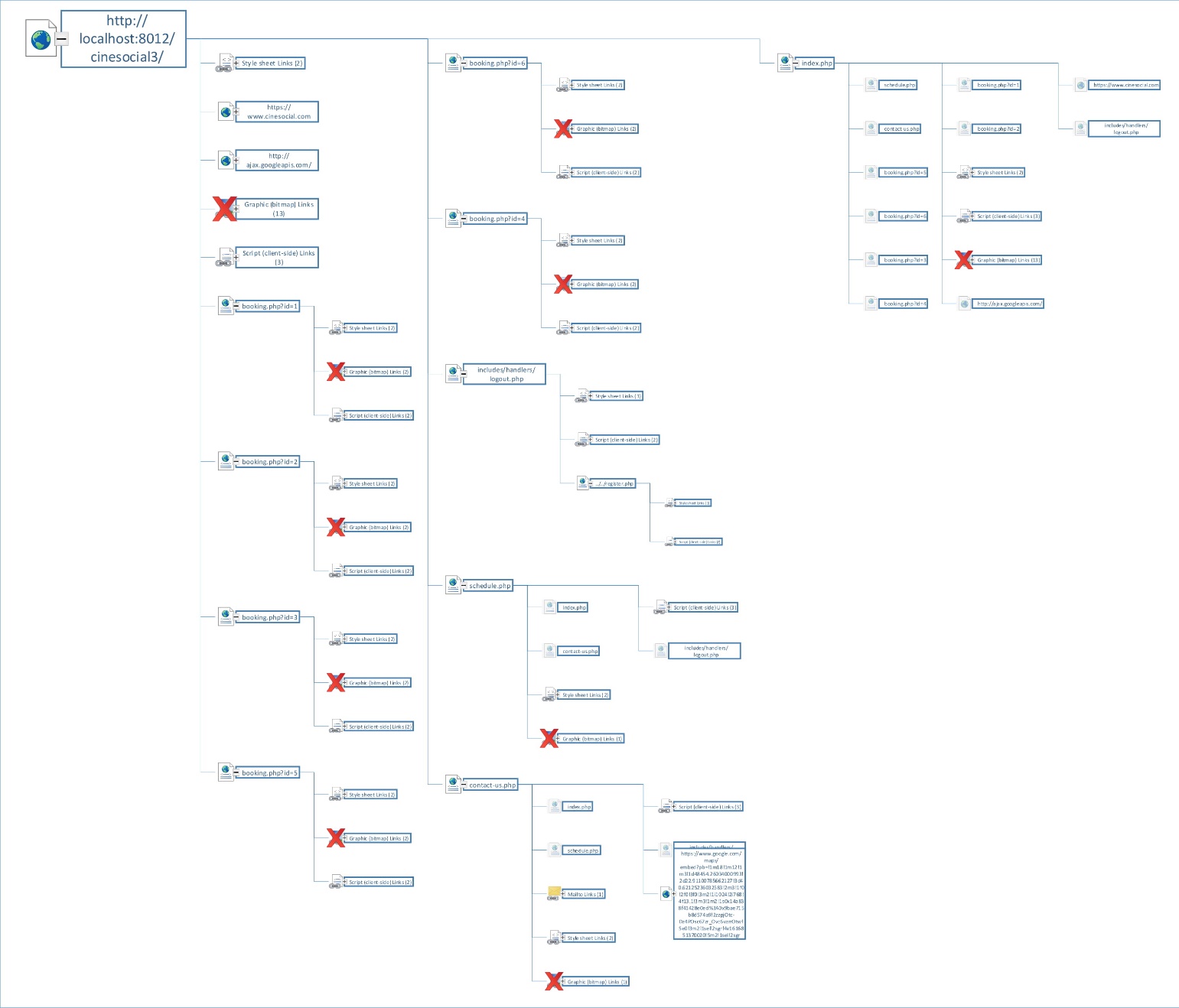
## **Storyboards & Wireframes (low fidelity)**

## **User Flows**

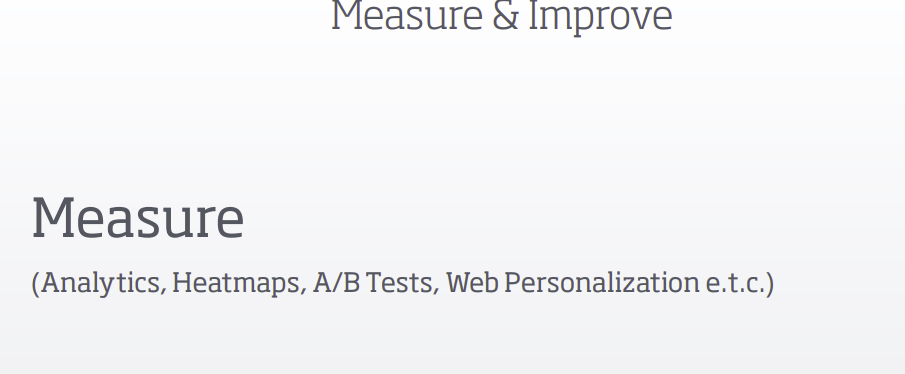


## **Web Site map Design & Planning**





## **Heat Map/mockup?**



# Project management