## **Software Requirement Specification**

Group members	Edris Afzali			
	Eva Jaghori			
	Neve McDonough			
	Haroon Almadani			
	Kaleb Henderson			
	Jonathan De La Pena			
	Zakirullah Amin			

#### 1. Project Description

DocDash is a mobile and web application designed to connect users with healthcare professionals, giving them the option to book appointments for scheduled visits or urgent care, all while being beneficiary to the convenience of the customers' homes. It provides an intuitive platform where the user can search for doctors based on their specialization, location, availability, and specifications. With various functionalities, DocDash improves the user experience, including filtering professionals, managing user profiles with personal and medical information, and communicating directly with doctors, via video meets and in-app messaging. The user will be able to pay for these services through secured payment gateways, along with using reviews, ratings, and doctor profile features.

#### 2. Functional Requirements

FR01	User Registration & Login: Users must be able to create an account and log in securely
FR02	<b>Profile Management:</b> Users can manage their profiles, including personal information, medical history, and contact details
FR03	<b>Doctor Search:</b> Users can search for available doctors based on specialty, location, and availability
FR04	Book Appointments: Users can book appointments for urgent or scheduled visits at home
FR05	Doctor Profiles: Display doctor credentials, ratings, reviews, and specialties
FR06	<b>Payment:</b> Users can pay for services through integrated payment gateways (credit card, insurance, etc.)
FR07	Appointment Cancellation/Rescheduling: Users can cancel or reschedule appointments based on doctor availability
FR08	<b>Doctor Ratings &amp; Reviews:</b> After an appointment, users can rate the doctor and leave feedback about their experience
FR09	<b>Doctor Specialization Filters:</b> Allow users to filter doctors based on specialization, such as pediatricians, cardiologists, or general practitioners
FR10	<b>Doctor Recommendations:</b> The app can suggest doctors based on the user's health history, location, or common conditions
FR11	<b>Appointment History:</b> Users can view their past appointments, including doctor details and visit notes
FR12	<b>Notification System:</b> Notify users of appointment confirmations, cancellations, and updates in real-time
FR13	<b>Emergency Protocols:</b> Provide an option for users to quickly contact emergency services if a doctor cannot arrive in time

FR14	Communication with Doctor: In-app messaging or video conferencing for pre-appointment		
	questions or consultations		
FR15	Multi-Language Support: Provide multi-language support for users who are non-native English		
	speakers		

## 3. Non-Functional Requirements

NFR01	Performance: The app should load within 2 seconds for 95% of operations.
NFR02	Scalability: The system must handle up to 1,000 concurrent users and doctors without
	performance degradation, with plans to scale this to 10,000 concurrent users within the next 12
	months.
NFR03	Security: Ensure all user data, including medical records and payment details, are encrypted
	using AES-256 and comply with HIPAA regulations. Conduct quarterly security audits and
	penetration tests.
NFR04	Availability: The system must maintain 99.9% uptime, equivalent to less than 8.76 hours of
	downtime per year
NFR05	Usability: The app interface should pass a System Usability Scale (SUS) test with a score of at
	least 80, ensuring ease of use, particularly for elderly or handicapped users.
NFR06	<b>Reliability:</b> The app must ensure that doctor arrival times are accurate within ±5 minutes in 90%
	of cases.

## 4. Use Case Specification

<< Select **three** functional requirements and describe them in detail using use cases.>>

UC01 Name:	User Registration & Login				
Description:	The app allows new users to create an account by providing their personal details and creating login credentials				
Actor:	User (Patient)				
Entry	The user selects the option "Sign Up" on the app's login screen.				
condition:					
Basic path:	<ol> <li>The system presents the registration form containing: [PRO02]         <ul> <li>a. Name (editable)</li> <li>b. Email (editable)</li> <li>c. Password (editable)</li> <li>d. Confirm Password (editable)</li> <li>e. Contact Number (editable)</li> <li>f. Address (editable)</li> <li>g. The options:                 <ul> <li>i. Submit</li> <li>ii. Go Back</li> </ul> </li> <li>The user fills in all required fields and selects the Submit option. [A01] [E02]</li> <li>The system verifies the input data for validity and uniqueness (e.g., email, password strength). [BR01] [BR02] [E01]</li> <li>The system creates the user account in the database.</li> <li>The system sends a confirmation email to the user with a verification link.</li> <li>The use case is concluded.</li> <li>The system returns to the login screen, prompting the user to log in.</li> </ul></li></ol>				
	[PRO01]				
Alternative paths:	<ol> <li>[A01] The user selects the Go Back option</li> <li>The use case is concluded.</li> <li>The system returns to the login screen. [PRO01]</li> </ol>				
Exception paths:	<ul> <li>[E01] The email provided is already registered</li> <li>1. The system displays an error message indicating that the email is already in use.</li> <li>2. The use case returns to step 1 of the basic path.</li> </ul>				

**[E02]** The user fails to fill in all required fields

- 1. The system displays an error message indicating which fields are missing.
- 2. The use case returns to step 1 of the basic path.

### Business Rules:

[BR01] All fields are mandatory

[BR02] The password must be at least 8 characters long and contain a mix of letters, numbers, and special characters

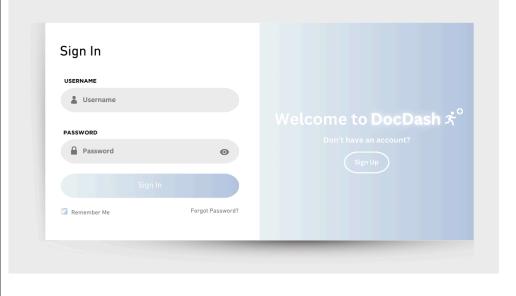
[BR03] The email must be unique and properly formatted

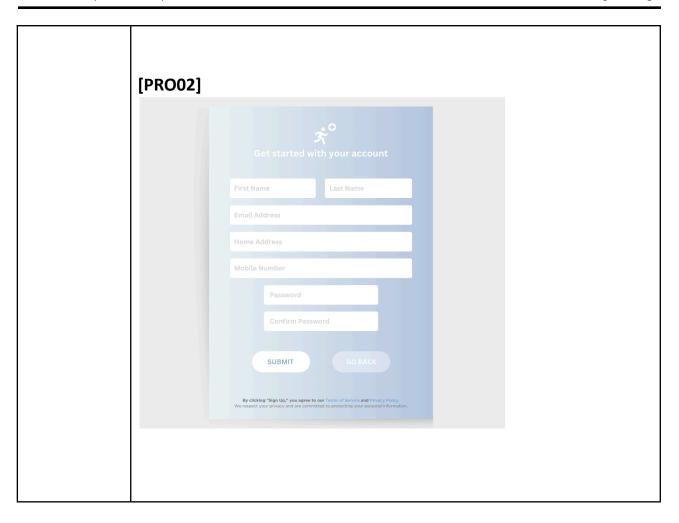
# Data description

Name	Туре	Length	Mask
Name	String	50	-
Email	String	100	formatted as a valid
			email address.
Password	String	MIN 8	contain a mix of
			letters, numbers,
			and special
			characters
Confirm	String	MIN 8	must match the
Password			Password field
Contact	String	10	#########
Number			
Address	String	100	-

### **Prototype:**

### [PRO01]





UC02 Name:	Payments				
Description:	The payment process ensures the user can securely pay using a selected method and receive confirmation that the payment was successful.				
Actor:	User (patient)				
Entry condition:	<ul> <li>The user reviewed their cart and clicked the "proceed to payment" then "checkout" button.</li> <li>The user must be logged into their profile to continue with the transaction.</li> </ul>				
Basic path:	1. The system presents the payment form [PRO01]  a. Present what kind of cards are accepted (VISA, Mastercard, etc.)  b. Payment due presented  c. Name on card (editable)  d. Card Number (editable)  e. Expiry date (mm/yy) (editable)  f. Security code (editable)  g. ZIP/Postal code (editable)  h. Options:  i. Pay amount  j. Apple Pay/Google Pay  k. Go back  2. The user fills in all required fields and selects the Make Payment option.  [A01] [A02]  3. The system verifies the input data for validity (i.e. the name, card number, CVC, expiry date, and ZIP/Postal code match) [BR01] [E02]  [E04]  4. The system processes the payment [E01] [E03] [A03]  5. The system displays a message asking if you would like to save this payment method to your profile [PRO02]  6. The system displays a payment complete message  7. The system sends an email to the user with confirmation of payment and option to seek a refund [BR02]  8. The use case is concluded.				
Alternative paths:	[A01] The user selects Apple Pay/Google Pay  1. The system takes in Apple Pay/Google Pay information				
	<ol> <li>The system takes in Apple Pay/Google Pay information</li> <li>The system verifies the input data for validity [BR01] [E02]</li> <li>The system processes the payment [A01] [E01] [E03]</li> <li>The system displays a payment complete message</li> <li>The system sends an email to the user with confirmation of payment and option to seek a refund [BR02]</li> <li>The use case is concluded.</li> </ol>				

#### [A02] User selects the go back button

1. The user is taken back to their cart

#### [A03] The payment is declined

- 1. The system displays a payment declined message and the option to choose another payment method
- 2. The original user payment information is wiped
- 3. The use case returns to step 1 of the basic path.

## Exception paths:

[E01] Payment gateway unavailable.

- 1. A retry logic mechanism takes forth, attempting to reconnect to the payment gateway after a brief delay
- 2. If the retry mechanism fails, an error message is displayed saying "We're sorry, but our payment processing service is temporarily unavailable. Please try again later."

[E02] Users' payment cards expired or invalid.

 Display an error message and prompts the user to "Edit Payment Information" or "Try Another Card"

**[E03]** Network lost, so transaction isn't completed.

- 1. A retry mechanism takes forth, attempting to retry the transaction after a brief pause
- 2. If the retry mechanism fails, an error message is displayed saying "Your transaction could not be completed because of a network error. Please check your internet connection and try again."

**[E04]** The user fails to fill in all required fields

- 1. The system displays an error message indicating which fields are missing.
- 2. The use case returns to step 1 of the basic path.

# Business Rules:

**[BR01]** System MUST accept only valid forms of supported payment methods. (e.g., debit and credit cards, PayPal, Apple and Google Pay, etc.)

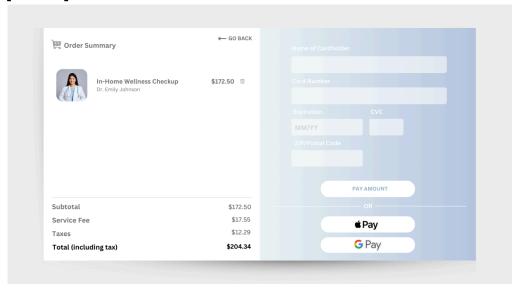
**[BR02]** Within 30 days (1 month) of original purchase date, the user may process a refund, but only within that time period, otherwise it is an invalid refund request.

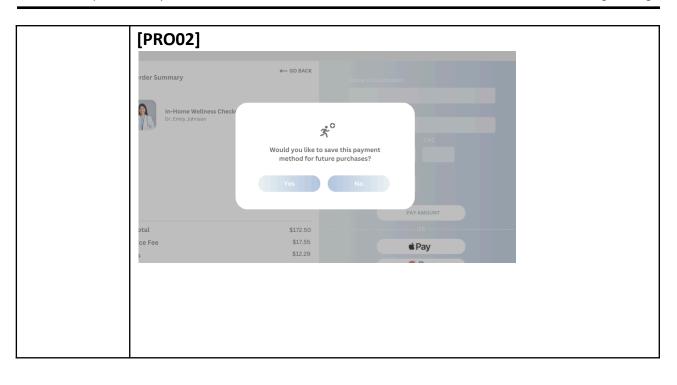
# Data description

Name	Туре	Length	Mask
Payment Due	Decimal	11	format as currency (e.g., \$###,###,###.##)
Name	String	50	No mask
Card number	String	16	#### #### #### ####
Expiry Date	String	5	MM/YY (e.g., ##/##)
CVC	String	3-4	### or #### depending on type of card
ZIP/Postal Code	String	5	#####

## **Prototype:**

## [PRO01]



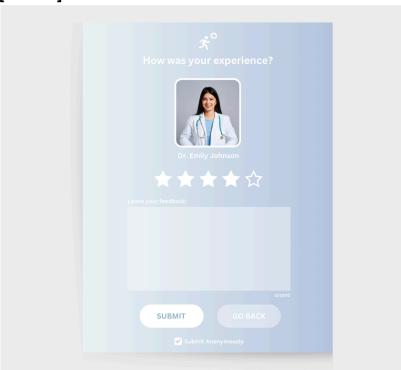


UC03 Name:	Doctor Ratings & Reviews				
Description:	After an appointment, users can rate the doctor and leave feedback about their experience with their doctor				
Actor:	User (patient)				
Entry condition:	<ul> <li>After the patient has completed an appointment, a "Review" button is available if they are logged into their account</li> <li>Additionally, after a patient has completed an appointment with their doctor, they receive an email asking the patient for a review, which will then direct them to the review page.</li> </ul>				
Basic path:	<ol> <li>The system presents a form to the user after an appointment [PRO01] [BR01][BR02] containing the following fields [E01][E02]:         <ul> <li>a. Doctor's name (non-editable)</li> <li>b. Rating (editable, scale 1-5)</li> <li>c. Feedback (editable, text area of up to 1000 characters)</li> <li>d. Submit Anonymously (optional checkbox)</li> <li>e. Options [A01]:</li></ul></li></ol>				

Alternative paths:	they conversely review  [A01] The us  1. The sy	historial histor	ry) Review (takes the e user back to the us appointment in not proceed with e user back to the	the user to their appointment user back to the review page) ir appointment history, where formation and their submitted a review (clicks "Go Back") ir appointment history, where formation and their submitted	i	
Exception	[ <b>E01</b> ] Connec	ction loss duri	ng review process			
paths:	1 A rotm	, machanism t	-akas farth attam	nting to reconnect		
	1. A retry mechanism takes forth, attempting to reconnect					
	<ol><li>If the retry mechanism fails, an error message is displayed saying "We encountered a network issue while trying to process your review.</li></ol>					
			ternet connection			
	<b>[E02]</b> The use	er doesn't fill (	out all the require	d fields		
	1. The system prompts the user to go back and fill out any empty fields					
	until the user either submits an appropriately filled out form or clicks "Go Back"					
	GO Back					
Business	[ <b>BR01</b> ] A rev	iew can only b	oe written AFTER	an appointment		
Rules:	[BR01] A review can only be written AFTER an appointment					
	[BR02] A user may write and leave a review for up to 30 days after an					
	appointment					
Data						
description	Name	Туре	Length	Mask		
	Review ID	Integer	10	########		
	Doctor	Integer	1	# (ranging from 1-5)		
	Rating					
	Feedback	String	Up to 1000	Any combination of		
			characters	characters and		
				symbols.		
	Anonymity	Boolean	1	Checked (true) or		
				unchecked (false)		

## **Prototype:**





## [PRO02]

